

Tutor Now



By:

Ahmed Abbas

38062

M.Ahmer Bilal Jan

30965

S.M Aliyan Rizvi

34822

Supervised by:

Mr. Osama Raza

**Faculty of Computing
Riphah International University, Islamabad
Spring 2025**

A Dissertation Submitted To

Faculty of Computing,

Riphah International University, Islamabad as a

Partial Fulfillment of the Requirement for the Award of

the Degree of

Bachelor of Science in Software Engineering

Faculty of Computing
Riphah International University, Islamabad
Date: 05/05/2025

Final Approval

This is to certify that we have read the report submitted by **Ahmed Abbas(38062)**, **M.Ahmer Bilal Jan(30965)**, **S.M Aliyan Rizvi(34822)**, for the partial fulfillment of the requirements for the degree of the Bachelor of Science in Software Engineering (BSSE). It is our judgment that this report is of sufficient standard to warrant its acceptance by Riphah International University, Islamabad for the degree of Bachelor of Science in Software Engineering (BSSE).

Committee:

1

Mr. Osama Raza
(Supervisor)

2

Dr. Musharaf Ahmad
(Head of Department/chairman)

Declaration

We hereby declare that this document “**Tutor Now**” neither as a whole nor as a part has been copied out from any source. It is further declared that we have done this project with the accompanied report entirely based on our personal efforts, under the proficient guidance of our teachers especially our supervisor **Mr. Osama Raza**. If any part of the system is proved to be copied out from any source or found to be reproduction of any project from anywhere else, we shall stand by the consequences.

Ahmed Abbas
38062

M.Ahmer Bilal Jan
30965

S.M Aliyan Rizvi
34822

Dedication

We dedicate this project to Allah Almighty our creator, our strong pillar, our source of inspiration, wisdom, knowledge and understanding. He has been the source of our strength throughout this program. Also, we dedicate our work to our family, friends and teachers. The unrivalled encouragement from our parents and outstanding support from teachers is what lead to success of this project. We also dedicate our work to our supervisor **Mr. Osama Raza** and the faculty members.

Acknowledgement

First of all, we are obliged to Allah Almighty the Merciful, the Beneficent and the source of all Knowledge, for granting us the courage and knowledge to complete this Project. We are grateful to our supervisor **Mr. Osama Raza** for their enthusiasm, patience, insightful comments, helpful information, practical advice, and unceasing ideas that always helped us tremendously in our project. Without his support and guidance, this project would not have been possible. Also, a special thanks to the rest of the faculty members for their unconditional support

Ahmed Abbas
38062

M.Ahmer Bilal Jan
30965

S.M Aliyan Rizvi
34822

Abstract

TutorNow is a cross-platform mobile application designed to connect students and parents with qualified tutors for personalized home tutoring sessions. The platform addresses key challenges in the home tutoring market, such as inefficient communication, lack of transparency, and the absence of progress tracking, by offering features like tutor profile verification, in-app messaging, and weekly progress reports. The project leverages Flutter for frontend development, Supabase for backend services, and GetX for state management, ensuring a scalable and user-friendly experience. Through a detailed analysis of existing systems, the team identified gaps and implemented solutions to streamline tutor discovery, enhance communication, and provide administrative oversight. The report outlines the project's functional and non-functional requirements, system design, implementation phases, and testing methodologies. Despite technical and adoption challenges, TutorNow successfully delivers a centralized, efficient, and transparent tutoring platform. The project demonstrates the potential to revolutionize the tutoring industry by fostering academic success and trust among stakeholders.

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction

1.1 Opportunity & Stakeholders

 1.1.1 Opportunity

 1.1.2 Stakeholders

1.2 Solution Overview

 1.2.1 Project Scope

 1.2.1.1 Tutor Profile Creation and Management

 1.2.1.2 Tutor Booking

 1.2.1.3 Tutor Availability Module

 1.2.1.4 In-App Communication Feature

 1.2.1.5 Progress Tracking

 1.2.1.6 Admin Dashboard

 1.2.1.7 Review and Feedback

1.3 Report Outline

Chapter 2: Literature/Market Survey

2.1 Introduction

2.2 Literature Review/Market Survey

 2.2.1 Survey of Existing Systems

 2.2.1.1 TeacherOn.com

 2.2.1.2 eTutors.pk

 2.2.1.3 TheTutors.pk

 2.2.1.4 Social Media Groups and Pages

 2.2.2 Existing Systems Comparison

2.3 Summary

Chapter 3: Requirement Analysis

3.1 Introduction

- 3.2 Problem Scenarios
- 3.3 Functional Requirements
- 3.4 Non-Functional Requirements

Chapter 4: System Design

- 4.1 Introduction
- 4.2 Detailed Design
 - 4.2.1 Architecture Diagram
 - 4.2.2 Use Case Diagram
- 4.3 Descriptive Use Cases
- 4.4 Activity Diagrams
- 4.5 Black Box Testing
- 4.6 Usability Testing
- 4.7 ERD Diagram

Chapter 5: Implementation

- 5.1 Work Breakdown Structure
- 5.2 Components, Libraries, and APIs
- 5.3 IDE, Tools, and Techniques
- 5.4 Deployment Diagram

Chapter 6: Conclusion and Outlook

- 6.1 Introduction
- 6.2 Achievements and Improvements
- 6.3 Critical Review
- 6.4 Future Recommendations and Outlook
- 6.5 Summary

List of Figures

4.2.1 Architectural Design

4.2.2 Use Case Diagram

4.4 Activity Diagrams

4.7 ERD Diagram

5.4 Deployment Diagram

List of Tables

2.2.2 Existing Systems

3.2 Problem Scenarios

4.3 Use Case Fully Dressed Format

4.5 Black Box Testing

4.6 Usability Testing

5.1 Work Breakdown Structure

Chapter 1:

Introduction

Chapter 1:

Introduction

TutorNow is a cross platform mobile application where students and parents are connected with qualified tutors. The application facilitates students or parents to book a home tutor for one-to-one personalized sessions. Additionally, the app also facilitates the parents to track students' progress all at one place. The application allows tutors to showcase their qualification, knowledge, skills, experience to help them standout in the crowded market of home tutoring. TutorNow aims to ensure quality tutoring service to students, create job opportunities for tutors and hassle-free progress monitoring of child for parents.

1.1 Opportunity and Stakeholders

1.1.1 Opportunities

- The home tutoring market in Pakistan is diverse, where many parents seek one-on-one sessions for their children. School systems often don't allow teachers to give individual attention to each student, creating a demand for personal tutors. This presents an opportunity for us to connect qualified tutors with students efficiently.
- The current method of finding a tutor is informal, relying heavily on word of mouth. Parents typically hire tutors based on personal recommendations. We can streamline this hiring process by offering a centralized platform where parents can easily find and evaluate tutors based on qualifications and reviews.
- Currently, communication between parents and tutors is inefficient, relying mostly on messaging apps. Tracking and maintaining records of a child's academic progress is often overlooked, leaving parents uninformed. We can bridge this communication gap by providing a more effective platform where progress is tracked, and regular updates are shared between tutors and parents.

1.1.2 Stakeholder

- Students
- Parents
- Tutors
- Admin

1.2 Solution Overview

Tutor Now will offer a mobile platform where tutors can create profiles showcasing their qualifications and experience, and parents/students can search, filter and book tutors based on specific needs such as name, subject, educational level, qualification, price range and availability. The app will include communication feature, progress tracking system that enables parents to monitor their child's academic performance. Admin functionalities will verify tutor profiles, manage users and resolve reports made by different users, while a review system will enhance trust.

1.2.1 Project Scope

1.2.1.1 Tutor Profile Creation and Management

- Tutors will be able to create and manage detailed profiles, which primarily include their qualifications, subjects offered, teaching experience, and availability.
- The admin will verify tutor profiles by reviewing their submitted documents, ensuring only qualified tutors can offer services on the platform.
- This verification process increases trust between tutors and users.
- Tutors will also be able to update their profiles and availability as needed to keep their offerings updated.

1.2.1.2 Tutor Booking

- Parents and students can search and filter various tutors based on name, subject expertise, qualification, and availability and price range.

- Students and parents can view the qualification and past teaching experience of tutors. They can also view the packages offered by tutors and the past feedback of tutors.
- This allows students and parents to quickly and effectively find tutors that meet their specific needs and make a booking.

1.2.1.3 Tutor Availability Module

- Tutors can set and manage weekly availability from Monday to Sunday by toggling days on or off and adding multiple custom time slots per day.
- Time slot validation ensures start times are before end times, maintaining the integrity of the schedule.
- Availability is automatically converted into bookable 30-minute slots for easy scheduling.
- Before confirming a session, the system checks for consecutive available slots matching the session duration and prevents double-booking by referencing existing bookings.
- The module supports multi-session or recurring bookings to accommodate long-term tutoring needs.
- Students can view only valid, conflict-free time slots in real time during the booking process.
- Time is displayed in both 12-hour and 24-hour formats for clarity and ease of use.
- This ensures smooth and efficient scheduling, avoiding overlaps and enhancing user experience for both tutors and students.

1.2.1.4 In-App Communication Feature

- A messaging system will be built into the app, enabling real-time communication between students, parents, and tutors and admin.
- This feature allows users to discuss tutoring needs, scheduling, and any specific academic challenges.
- By keeping communication within the app, the platform ensures all interactions are trackable and secure, reducing the likelihood of disputes.

1.2.1.5 Progress Tracking

- Tutors will be required to submit progress reports on students' academic performance through the app.
- Parents and students will have access to this data via their profiles, making it easy to monitor learning outcomes.
- The progress tracking system will also feature weekly summaries, which can help parents stay informed and support their child's education more effectively.

1.2.1.6 Admin Panel

- The admin panel will include a dashboard where admin can view the essential analytics such as platform engagement, profile requests, booking requests, graphical analysis of tutors qualification and experience and filtering and viewing the recent bookings made on Tutor Now..
- The tutors will be verified based on the documents uploaded by them which includes personal details, qualification details and experience details.
- Users can submit reports against other users by providing details with proofs in the form of images which are then reviewed and resolved by admin by taking appropriate actions through the admin panel.
- Admin can also block and unblock users, and admin can communicate with any user.

1.2.1.7 Review and Feedback

- After completing a tutoring session or program, parents and students will be able to rate tutors on criteria such as teaching quality, punctuality, and communication etc.
- To ensure fairness, users will be able to leave feedback, promoting a balanced review system.
- This feedback will be visible on tutor profiles to help future users make informed decisions.

1.3 Report Outline

This report covers the detail of all aspects of the system, for understanding and clarity.

This report has been divided into seven chapters.

1.3.1 Chapter 1

This chapter 01 includes the opportunities which were initially created for you which gave us the motivation and opportunity to look more deeper into the ideation of tutoring market.

1.3.2 Chapter 2

This chapter focuses on the current situation of the market, and what are the pros and cons of the various existing systems.

1.3.3 Chapter 3

This chapter is all about problems identified by deep and detailed analysis of existing systems and then its includes specifying the functional and non-functional requirements .

1.3.4 Chapter 4

This chapter provides all the information related to design factor of the developed system by describing the system architecture design consideration and different diagrams that model the working behaviour of the system.

1.3.5 Chapter 5

This chapter provides information about the work breakdown structure (WBS) and then it highlights the tools and technologies used during the implementation of the project.

Chapter 2:

Literature/Market Survey

Chapter: 2

Literature/Market Survey

2.1 Introduction

In this chapter, we are discussing the literature/market survey of our project

“Tutor Now”. For our project, we have used interviewing technique to gather the functional requirements of the project. Further, we have created a table in which we have compared the features of the similar existing systems of our project. We critically analysed those existing systems and identified the pros and cons of those systems to better understand their current flow of work and identify gaps.

2.2 Literature Review/Market Survey

The **Literature Review/Market Survey** includes the survey of existing systems which are our competitors in the market. Those existing system are being operated online. Those systems include Teacheron.com, etutors.pk, thetutors.pk and social media groups and pages. Individually each of these systems have been discussed below and then a comparative analysis of these system has been shown in a table based on various features.

2.2.1 Survey of Existing Systems

2.2.1.1 Teacheron.com

TeacherOn provides a place for students and tutors to meet. TeacherOn provides:

1. Online teaching service
2. Home tutoring service
3. Assignment help service

There business model is based on coins, and they charge if payment is made through their platform.

How it works - Students

Students register themselves on website and they get two options; one is to post a specific requirement or view the posts of tutors. TeacherOn provides students 150 free coins once they register themselves but later you must purchase. These coins are required to message or call a tutor. Students can pay directly to tutors or via TeacherOn.

How it works - Tutors

Tutors register and complete profile details to contact students. TeacherOn verifies profile based on documents. As a tutor, it's free if a student contacts you; contacting students may require coins, with costs ranging from 0 to 100+ coins. Tutor decides the coins required by a student to contact him.

2.2.1.2 etutors.pk

eTutors.pk is online tutor search portal which pairs students and tutors for home/online tuition lessons. There business model is based on taking a %age of agreed upon tuition charges from tutors.

How it works - students

Students register themselves and then students can either search for a tutor or request a tutor. Student sends a hiring request to tutor and then eTutors.pk connects you based on availability. Request a tutor option provides you to enter your requirements and then system matches suitable tutor using algorithm.

How it works - Tutors

Tutors register and complete profile details to contact students. eTutors.pk verifies profile based on documents. Tutors appear on top of search results based on tutors' points. A

student can give a tutor up to 50 feedback points, plus 5 points for a helpful answer. There are three levels of tutors:

1. Beginner (less than 500 points)
2. Intermediate (between 500 to 1000 points)
3. Professional (greater than 1000 points)

Tutors can apply for jobs in recent jobs option. eTutors.pk offers two types of tuition:

1. Regular tuition (local students)
2. Premium tuition (abroad students)

For regular tuition, eTutors Academy charges 50% of the first month's fee and pays the tutor after three weeks of successful tuition. For Premium Tuitions, eTutors Academy charges 40% of the first month's fee and 20% of subsequent months' fees.

2.2.1.3 thetutors.pk

TheTutors.pk helps students find home tutors and provides a platform for parents to choose tutors based on their needs.

How it works – Students

Students find qualified tutors in their area by selecting your location, class, and subjects. After submitting your request, the coordinator contacts students within 30 minutes on WhatsApp to discuss details. A 2-day free trial is available, and the fee is taken in advance after the trial period. TheTutors offers a variety of courses, including IELTS, TOEFL, CSS, MCAT, ECAT, and entry test preparation.

How it works – Tutors

Tutors register by filling out a form with their subjects, classes, fee, and preferred areas. Online tutors select the online tutoring option during registration. Verification requires sending NIC images via WhatsApp. Tutors can add a short introduction video to enhance

the profile. Jobs are typically assigned within 1-2 months, depending on the profile. Thetutors.pk charges 50% of the first month's fee and a 500-registration fee.

2.2.1.4 Social Media Groups and pages

Students and tutors are connected across various social media platforms such as Facebook and Instagram. In groups on Facebook, the students and tutors communicate through posts. A student posts about his/her requirement whereas the tutor might mention about his/her availability to teach a particular subject. Whereas on pages, local agencies post advertisements which include information about tutors and contact numbers are mentioned.

2.2.2 Existing Systems

Features	teacheron.com	eTutors.pk	thetutors.pk	Social Media Groups and pages
Direct Communication with Tutor	☒	✗	☒	☒
Verified Tutor Profiles	☒	☒	✗	✗
Percentage Fee on Tuition Charges	✗	☒	☒	✗
Feedback Rating System	☒	☒	✗	✗
Tutor Levels Based on Points	✗	☒	✗	✗
Request a Tutor Feature	✗	☒	☒	✗
Free Of Cost Communication	✗	☒	☒	☒

Table 2.3.1 | Existing Systems

2.3 Summary

From our review of existing systems, we found that none provide a complete solution that ensures clear communication, timely updates, and transparency for all users. Most platforms are decentralized, with communication happening outside the application, searching and filtering criteria is limited, tutors profile verification criteria is not really effective and efficient and no tracking of student progress is involved. This is why we came up with the idea of Tutor Now. It offers a one-stop solution for connecting students, parents and tutors with features like easy messaging, effective booking based on your requirements and progress reports. Tutor Now focuses on providing clear communication and quality service, making it the best choice for students, tutors, and parents.

Chapter 3:

Requirement Analysis

Chapter 3:

Requirement Analysis

3.1 Introduction

In this chapter, we will discuss the Functional & Non-Functional requirements of our project “**Tutor Now**”. Prior to that, we will discuss all the problem statements we have found while doing research on the project idea. These functional requirements are gathered from the client using several techniques like brainstorming, observation and study of existing systems. The Non-Functional requirements are gathered by observing the functional requirements.

3.2 Problem Scenarios

The problem statement - for students, parents and tutors	
The problem of	lack of centralized communication
Affects	students, parents and tutors
The result of which	There is delay and inefficiency in sharing information about students' progress.
Benefits of	Improved communication and timely updates, leading to better academic support and coordination

Table 3.2.1 | Problem Statement 1

The problem statement - for System	
The problem of	Direct communication outside the system.
Affects	System
The result of which	The platform may lose users and eventually lose revenue.
Benefits of	Increased user retention and revenue through enforced communication within the app.

Table 3.2.2 | Problem Statement 2

3.3 Functional Requirements

3.3.1 Student/Parent Functional Requirements:

1. The student/parent shall be able to register an account by selecting user type and entering full name, email, password, confirm password.
2. The system shall send a verification link on the entered email of the student/parent.
3. The student/parent shall be able to complete the registration process by clicking on the link sent on their email.
4. The student/parent shall be able to login to the account by entering his/her email and password.
5. Student/Parent shall be able to complete their profile by entering their details which includes uploading their picture, educational level, city, subjects I am weak at and learning goals.
6. The student/parent shall be able to edit their profile.
7. The student/parent shall be able to search the tutors and filter the tutors based on subject, name, education level, qualification and price level.
8. The system shall allow student/parent to view tutor profiles.

9. Students/Parents shall be able to send messages to tutors.
10. Students/Parents shall be able to view incoming messages from tutors.
11. The student/parent shall be able to book the tutor based on availability.
12. Student/Parent shall be able to provide feedback on completed bookings.
13. Student/Parent shall be able to rate tutors after sessions on the scale of 1-5 where 1 is lowest rating and 5 is highest rating.
14. Student/Parent shall be able to view past ratings and feedback provided to a tutor.
15. Student/Parent shall be able to report issues or disputes with tutors within the app.
16. Student/Parent shall be able to view weekly progress reports provided by tutors.
17. Students/Parents shall be able to archive chats.
18. Students/Parents shall be able to unarchive chats.
19. Students/Parents shall be able to pin chats.
20. Students/Parents shall be able to unpin chats.

3.3.2 Tutor Functional Requirements:

1. The tutor shall be able to register an account by selecting user type and entering full name, email, password, confirm password.
2. The system will require the tutor to enter a strong password with at least 8 characters, at least one uppercase character, at least one number, and at least one special character.
3. The system shall send a verification link on the entered email of the tutor.
4. The tutor shall be able to complete the registration process by clicking on the link sent on their email.
5. The tutor shall be able to login to the account by entering his/her email and password.
6. The system will require tutors to enter their details which includes adding bio, location, qualifications, past experience and subjects offered.

7. The tutor must upload his/her image, original CNIC, qualification and experience documents in pdf format for verification.
8. The tutors shall be able to update their profile.
9. Tutors shall be able to add and update their availability schedule.
10. Tutors shall be able to create packages which includes name, description, duration of each session, number weeks, total number of sessions and price.
11. Tutors shall be able to view incoming messages from students/parents.
12. Tutors shall be able to send messages to students/parents.
13. Tutors shall be able to accept or decline bookings.
14. Tutors shall be able to cancel bookings .
15. Tutors shall be able to generate progress reports for students.
16. Tutors shall be able to report disputes with student/parent within the app.

3.3.3 Admin Functional Requirements:

1. Admin shall be able to verify the personal, qualification, experience details and documents of tutors.
2. Admin shall be able to approve tutor profiles.
3. Admin shall be able to reject tutor profiles.
4. Admin shall be able to resolve reports submitted by tutors and student/parent.
5. Admin shall be able to communicate with parents, students and tutors through in app messaging feature.
6. Admin shall be able to block users
7. Admin shall be able to unblock users.
8. Admin shall be able to view rejection history of tutors.

3.4 Non Functional Requirements

The non-functional requirements include:

1. Usability:

- The user interface shall be intuitive.
- The app shall have a simple, navigation system with clear labels for all features.
- The app shall use consistent design patterns, such as button placement, colour schemes, and typography, to ensure familiarity across all screens.
- Tutors shall be shown profile completion status, encouraging them to finish all required sections.

2. Adaptability:

- The system must enable the seamless addition of new features or components to any layer (Model, View, Controller) without impacting existing functionality.

3. Maintainability:

- The application must ensure a clear separation of concerns between the Model, View, and Controller layers, enabling independent updates and troubleshooting of each layer.
- The system must allow team members to work on different layers simultaneously with minimal risk of conflicts, promoting efficient development and maintenance.

.

Chapter 4:

System Design

Chapter 4:

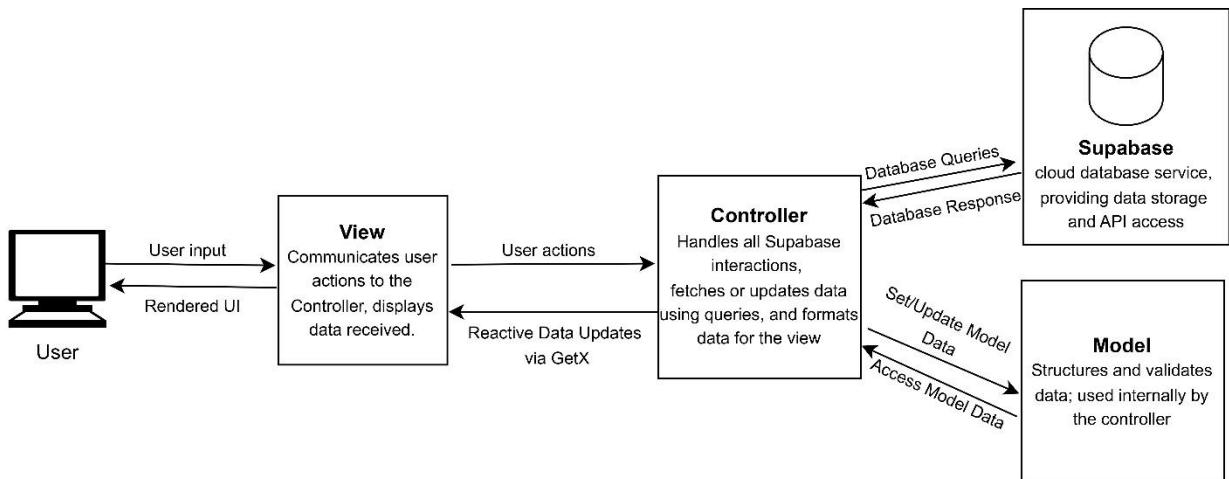
System Design

4.1 Introduction

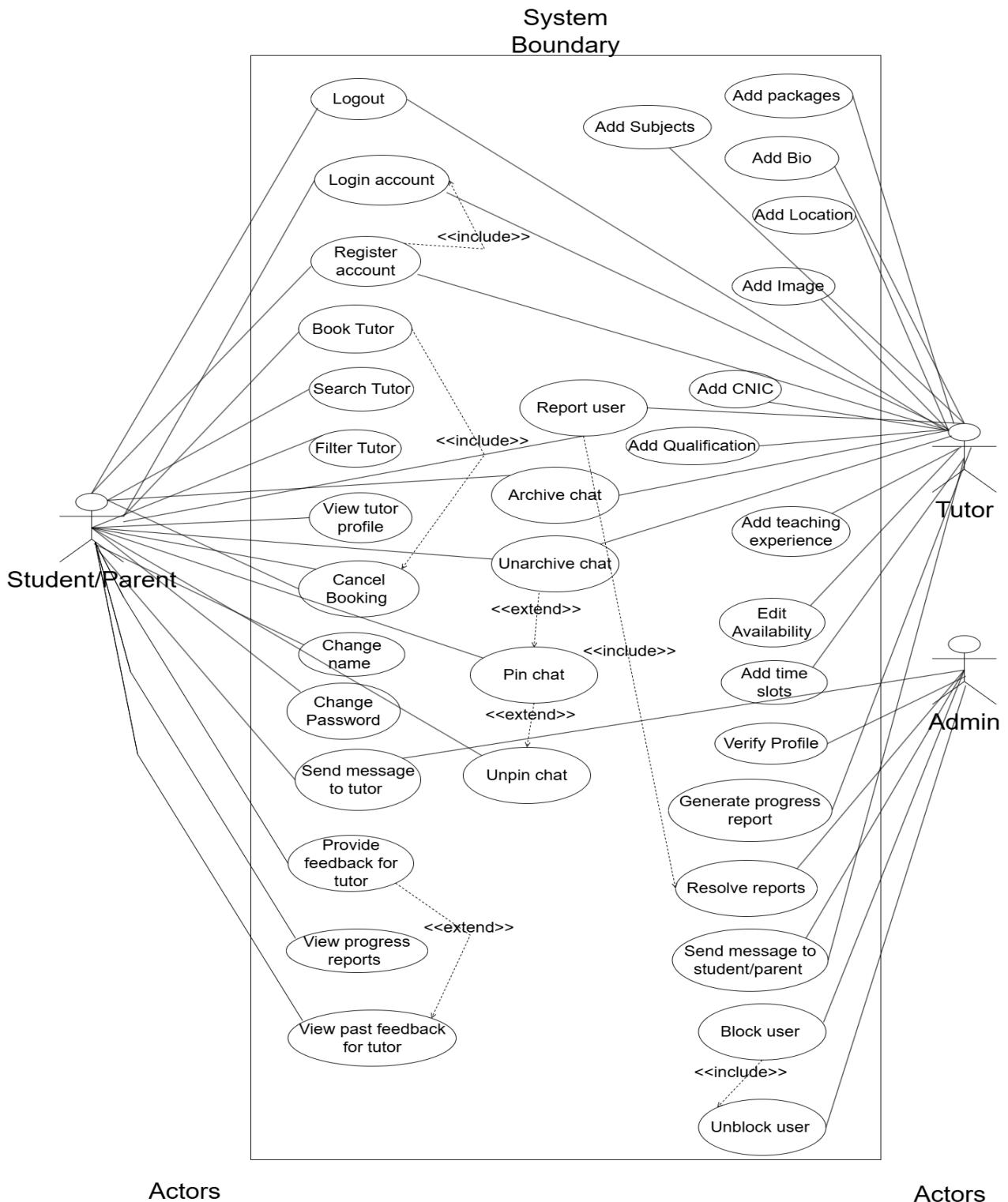
In this chapter of “System design”, architectural and detailed design of the system is shown. We have initially identified all the use cases from the gathered functional requirements. Then each use case is descriptively explained using the fully dressed use case to clearly illustrate the actors involved, preconditions and post conditions and the users’ actions and the system response against the user actions. Then to better explain the flow of each use case activity diagram are drawn. For showing how user interacted with frontend and then how UI interacted with database, sequence diagrams are drawn. At the end of this chapter black box testing has been performed in detail.

4.2 Detailed Design

4.2.1 Architecture Diagram



4.2.2 Use Case Diagram



4.3 Descriptive Use Cases (Fully Dressed Use cases)

UC-ID#	1	
Use Case Name	Login Account	
Primary Actor	Student, Parent, Tutor, Admin	
Pre-condition	The user has registered and verified the email by clicking the link sent to their registered mail.	
Post-condition	The user is successfully logged in to their account	
Normal Flow	User Action: 1. Enter email. 2. Enter password. 3. Click login button.	System Response: 4. System verified the credentials. 5. Upon successful login, user is redirected to profile screen.
Alternate Flow	A5. If the user enters wrong email or password, system displays an error message showing “Invalid email or password” and prompts the user to try again.	

UC-ID#	2	
Use Case Name	Register Account	
Primary Actor	Student, Parent, Tutor	
Pre-condition	The user has access to valid email address.	
Post-condition	The user has successfully registered and is ready to login.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. User first selects user type. 2. User enters email. 3. User enters name. 4. User enters password. 5. User enters confirm password. 6. User clicks register button. 	<p>System Response:</p> <ol style="list-style-type: none"> 7. System validates all the fields. 8. System sends verification link to entered email and shows successful registration message upon email confirmation.
Alternate Flow	<p>A8.1 If the user enters an invalid email, then the system displays an error message saying, "Enter a valid email".</p> <p>A8.2 If the user enters a name that is less than 2 characters or greater than 50 characters, then the system displays an error message asking the user to "Enter a valid name".</p> <p>A8.3 If the user's password does not meet the required password criteria, then the system displays an error message and prompts the user to "Fulfil the password requirements".</p> <p>A8.4 If the password and confirm password fields do not match, then the system prompts the user to "Enter the password and confirm password again".</p> <p>A8.5 If any required field or selection is missing, then the system displays an error message asking the user to "Fill in all the fields".</p>	

UC-ID#	3	
Use Case Name	Add Bio	
Primary Actor	Tutor	
Pre-condition	User has successfully logged in.	
Post-condition	User's bio is successfully saved, and the user is redirected to the next incomplete step.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. User enters a valid bio into the text field (1–150 characters). 2. User clicks the "Next" button. 	<p>System Response:</p> <ol style="list-style-type: none"> 3. System validates the bio input 4. System saves the bio to database and redirects the user to the next incomplete step.
Alternate Flow	<p>A4.1 System displays an error message: "Bio field is empty."</p> <p>A4.2 System displays an error message: "Bio must not exceed 150 characters."</p>	

UC-ID#	4	
Use Case Name	Add Location	
Primary Actor	Tutor	
Pre-condition	User has successfully logged in.	
Post-condition	User's location is successfully saved, and the user is redirected to the next incomplete step.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. User selects a Province from the dropdown menu. 2. User selects a city from the dropdown menu. 3. User clicks the "Next" button. 	<p>System Response:</p> <ol style="list-style-type: none"> 3. System validates the location input. 4. System saves the location to database and redirects the user to the next incomplete step.
Alternate Flow	<p>A4.1 System displays an error message: "Province not selected."</p> <p>A4.2 System displays an error message: "City not selected"</p>	

UC-ID#	5	
Use Case Name	Add Image	
Primary Actor	Tutor	
Pre-condition	User has successfully logged in.	
Post-condition	User's image is successfully saved, and the user is redirected to the next incomplete step.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. User clicks on tap to upload. 3. User selects an image file. 4. User clicks the "Next" button. 	<p>System Response:</p> <ol style="list-style-type: none"> 2. The system opens the gallery for the user to select an image. 5. System validates the selected image. 6. System saves the image and redirects the user to the next incomplete step.
Alternate Flow	<p>A6.1 If the user does not select an image, system prompts to select an image.</p> <p>A6.2 If the user selects an invalid image type, system prompts to select a valid image type.</p> <p>A6.3 If the user selects an invalid image size, system prompts to select a valid image size.</p>	

UC-ID#	6	
Use Case Name	Add CNIC	
Primary Actor	Tutor	
Pre-condition	User has successfully logged in.	
Post-condition	User's cnic is successfully saved, and the user is redirected to the next incomplete step.	
Normal Flow	<p>User Action</p> <ol style="list-style-type: none"> 1. User clicks on tap to upload. 3. User selects a file. 4. User clicks the "Next" button. 	<p>System Response:</p> <ol style="list-style-type: none"> 2. The system opens the file manager for the user to select a file. 5. System validates the selected file. 6. System saves the file and redirects the user to the next incomplete step.
Alternate Flow	<p>A6.1 If the user does not select any file, system prompts to select a file.</p> <p>A6.2 If the user selects an invalid file type, system prompts to select a valid file type.</p> <p>A6.3 If the user selects an invalid file size, system prompts to select a valid file size.</p>	

UC-ID#	7	
Use Case Name	Add Qualification	
Primary Actor	Tutor	
Pre-condition	User has successfully logged in.	
Post-condition	User's qualification is successfully saved, and the user is redirected to the next incomplete step.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. User selects the education level from dropdown menu. 2. User enters the institute name. 3. User clicks on tap to upload to upload qualification document. 5. User selects a file. 6. User clicks the "Next" button. 	<p>System Response:</p> <ol style="list-style-type: none"> 4. The system opens the file manager for the user to select a file. 7. System validates the input fields. 8. System saves the details and file and redirects the user to the next incomplete step.
Alternate Flow	<p>A8.1 If the user doesn't select educational level, the system prompts user to select educational level</p> <p>A8.2 If the user doesn't enter institute name, the system prompts user to enter institute name.</p> <p>A8.3 If the user does not select an image, system prompts to select an image.</p> <p>A8.4 If the user selects an invalid file type, system prompts to select a valid file type.</p> <p>A8.5 If the user selects an invalid file size, system prompts to select a valid file size.</p>	

UC-ID#	8	
Use Case Name	Add Teaching Experience	
Primary Actor	Tutor	
Pre-condition	User has successfully logged in.	
Post-condition	User's teaching experience is successfully saved, and the user is redirected to set availability.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. User selects the education level of students from dropdown menu. 2. User selects the start date of their teaching experience. 3. User selects the end date of their teaching experience or toggles "I still work here." 4. User clicks on tap to upload to upload experience document. 6. User selects a file. 7. User clicks the "Next" button. 	<p>System Response:</p> <ol style="list-style-type: none"> 5. The system opens the file manager for the user to select a file. 8. System validates the input fields. 9. System saves the details and file and redirects the user to set up availability.
Alternate Flow	<p>A9.1 If the user doesn't enter educational level of students, the system prompts user to select educational level of students</p> <p>A9.2 If the user doesn't select start date, the system prompts user to enter start date.</p> <p>A9.3 If "I still work here" is toggled off but the end date is not selected the system prompts user to select an end date.</p> <p>A9.4 If the user does not select any file, system prompts them to select a file.</p> <p>A9.5 If the user selects an invalid file type, system prompts to select a valid file type.</p>	

	A9.6 If the user selects an invalid file size, system prompts to select a valid file size.
--	---

UC-ID#	9	
Use Case Name	Edit Profile	
Primary Actor	Student, Parent, Tutor	
Pre-condition	The user has logged in their account	
Post-condition	User profile is updated and displayed on profile screen.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. User navigates to the Profile screen. 2. User clicks on the Edit Profile button. 4. User updates the respective profile detail. 5. User clicks the Save button to confirm changes. 	<p>System Response:</p> <ol style="list-style-type: none"> 3. System displays a form allowing the user to edit their profile details. 6. System validates the input. 7. System updates the profile details. 8. System displays the updated profile details on the profile screen.
Alternate Flow	A7.1 If system fails to update profile details, system displays an error message “Failed to update profile details try again”.	

UC-ID#	10	
Use Case Name	Change Password	
Primary Actor	Student, Parent, Tutor	
Pre-condition	The user has logged into their account	
Post-condition	User password is updated	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. User navigates to profile screen. 2. User Selects the edit option. 3. User selects change password option. 4. User enters old password. 5. User enters new password. 6. User confirms new password. 7. User clicks save button. 	<p>System Response:</p> <ol style="list-style-type: none"> 8. System validates all fields 9. User password changed successfully.
Alternate Flow	<p>A9.1 If user enters incorrect old password, system displays an error message and prompts user to “Enter password again”.</p> <p>A9.2 If user enters a weak password, system displays error message and prompts the user to “Meet minimum password requirements”.</p> <p>A9.3 If confirm new password doesn’t match new password, system prompt with error message to “Match password and confirm password field”.</p>	

UC-ID#	11	
Use Case Name	Verify tutor profile	
Primary Actor	Admin	
Pre-condition	The tutor has completed profile and uploaded required documents.	
Post-condition	Profile is either approved or rejected.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. Admin clicks approve tutors in the side navigation bar. 3. Admin clicks on view detail button for the profile which he wants to verify. 5. Admin clicks approve tutor button if requirements are met. 	<p>System Response:</p> <ol style="list-style-type: none"> 2. System displays list of all tutors with name, qualification and current status. 4. The system displays the personal, qualification and experience details. 6. System successfully approves the tutor showing “Tutor approved successfully”.
Alternate Flow	<p>A6. If profile requirements are not met, then admin clicks reject button, and system prompts admin to select the rejection reason. Admin selects rejection reason and click reject profile button which displays “Admin rejected successfully”</p>	

UC-ID#	12	
Use Case Name	Search Tutor	
Primary Actor	Student, Parent	
Pre-condition	The Student/Parent is logged in to the application.	
Post-condition	List of tutors matching the search criteria is displayed.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. The Student/Parent navigates to the tutor search page by clicking search in navigation bar. 2. The Student/Parent search tutor based on name, subject and educational level. 3. The student/parent click on search button. 	<p>System Response:</p> <ol style="list-style-type: none"> 4. The system processes the search. 5. The system displays a list of tutors that matches the criteria.
Alternate Flow	A5. If no tutors match the search criteria, the system displays a message indicating no tutors found.	

UC-ID#	13	
Use Case Name	Filter Tutor	
Primary Actor	Student, Parent	
Pre-condition	The Student/Parent are logged in and have navigated to the tutor search page.	
Post-condition	List of tutors matching the filter criteria is displayed.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. The student/parent navigate to search page. 2. The Student/Parent selects filter options which include educational level, qualification, price range. 	<p>System Response:</p> <ol style="list-style-type: none"> 3. The system applies the filters to the available tutors list.

		4. The system displays a filtered list of tutors.
Alternate Flow	A4. If no tutors match the filter criteria, the system displays a message indicating no tutors found.	

UC-ID#	14	
Use Case Name	View Tutor Profile	
Primary Actor	Student, Parent	
Pre-condition	The Student/Parent has accessed the list of tutors and selected a tutor to view their profile.	
Post-condition	Tutor profile is displayed	
Normal Flow	User Action: 1. The Student/Parent selects a tutor from the tutors list.	System Response: 2. The system retrieves the tutor's profile information. 3. The system displays the tutor's profile to the Student/Parent.
Alternate Flow	-	

UC-ID#	15	
Use Case Name	Book Tutor	
Primary Actor	Student, Parent, Tutor	
Pre-condition	Student/Parent must be logged in and the selected tutor must have availability.	
Post-condition	Booking request is sent to the tutor.	
Normal Flow	User Action: 1. The Student/Parent clicks on a tutor profile. 3. The student/parent views profile.	System Response: 2. The system displays tutor profile.

	<p>4. The student/parent selects the package.</p> <p>5. The Student/Parent selects a time slot for each session by clicking next button.</p> <p>6. The student/parent confirms the booking by clicking confirm booking button after selecting the slot for last session.</p> <p>8. The tutor receives the booking request and either accepts or declines.</p>	
Alternate Flow		<p>7. The system sends a booking request to tutor.</p> <p>9. If accepted, the system displays the booking in active bookings section to student/parent.</p> <p>A9. If the booking request was declined, the system displays the booking in declined bookings section to student/parent.</p>

UC-ID#	16	
Use Case Name	Edit Availability	
Primary Actor	Tutor	
Pre-condition	Tutor must be logged in	
Post-condition	Tutor's availability is updated	
Normal Flow	User Action: 1. The tutor goes to the set availability screen. 2. The tutor selects edit availability option. 3. The tutor edits time slots and saves them.	System Response:

		4. The system updates the availability.
Alternate Flow	A4. If start time is later than the end time or vice versa then error message is displayed.	

UC-ID#	17	
Use Case Name	Add time slots	
Primary Actor	Tutor	
Pre-condition	Tutor must be logged in	
Post-condition	Tutor's availability is updated	
Normal Flow	User Action: 1. The tutor goes to the set availability screen. 2. The tutor clicks edit availability button. 3. The tutor clicks add availability button to add time slot to a day.	System Response: 4. The system adds a time slot to that day.
Alternate Flow	-	

UC-ID#	18	
Use Case Name	Send Message to tutor	
Primary Actor	Student, Parent	
Pre-condition	The user must be logged into the system.	
Post-condition	The message has been successfully sent to the tutor.	
Normal Flow	User Action: 1. The student navigates to a tutor's profile or chat list screen. 2. The student clicks the "Chat" button	System Response:

	<p>in tutor profile or opens the chat from chat list screen which opens the chat screen.</p> <p>4. The student types the message content in the input field.</p> <p>5. The student clicks the "Send" button.</p> <p>7. The tutor receives the message.</p> <p>8. The tutor opens the chat to read and respond.</p>	<p>3. The system displays the chat interface with the tutor.</p> <p>6. The system processes the request and sends the message in real-time.</p>
Alternate Flow	A6. If the system fails to send the message, an error message is displayed to try again .	

UC-ID#	19	
Use Case Name	Send Message to Student/Parent	
Primary Actor	Tutor	
Pre-condition	The tutor must be logged into the system as a verified tutor by Tutor Now.	
Post-condition	The message is successfully sent to the student/parent.	
Normal Flow	User Action: 1. The tutor selects the student/parent and clicks that	System Response:

	<p>chat to send a message to in chat list screen.</p> <p>3. The tutor types the message content in the input field.</p> <p>4. The tutor clicks the "Send" button.</p> <p>6. The student receives the message.</p> <p>7. The student opens the chat to read and respond.</p>	<p>2. System display the chat interface.</p> <p>5. The system processes the request and sends the message in real-time.</p>
Alternate Flow	A5. If the system fails to send the message, an error message is displayed.	

UC-ID#	20	
Use Case Name	Generate Weekly Progress Reports	
Primary Actor	Tutor	
Pre-condition	The tutor has accepted the booking request and the booking displays in active bookings.	
Post-condition	Weekly progress report is saved in database and is visible to student.	
Normal Flow	User Action: 1. Tutor clicks on the active booking for which he want to generate progress report.	System Response: 2. System displays about and progress tabs.

	<p>3. The tutor clicks on the progress tab.</p> <p>5. The tutor clicks on the week for which he wants to generate progress report.</p> <p>7. Tutor enter the overall performance and additional comments for that week.</p> <p>8. The tutor clicks on save button.</p> <p>10. The tutor clicks on attach images button.</p> <p>12. The tutor taps to upload images and enters comments to images.</p> <p>13. Tutor clicks on save button.</p>	<p>4. System shows the total number of weeks for package.</p> <p>6. System displays a form to select overall performance and enter additional comments.</p> <p>9. System saves the report to database.</p> <p>11. System navigates to attach images screen.</p> <p>14. System saves the progress report images of a specific week in database.</p>
Alternate Flow	A9. If the system fails to save the report, an error message is displayed “Failed to save report try again”.	

	A14. If system fail to save the progress report images and comments, an error message is displayed showing “Failed to save report images try again ”.
--	---

UC-ID#	21	
Use Case Name	View Weekly Progress Reports	
Primary Actor	Student, Parent	
Pre-condition	The student is logged in to the app and progress reports are available for the week which is to be viewed.	
Post-condition	The student/parent views progress reports.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. The student/parent clicks on bookings in navigation bar. 3. The student/parent clicks on specific booking in active bookings section for which the progress reports are to be viewed. 5. The student clicks on progress tab. 7. The student/parent clicks on the specific week. 	<p>System Response:</p> <ol style="list-style-type: none"> 2. The system displays bookings 4. The system display progress and about tabs. 6. The system retrieves the current overall performance and weeks for which the progress reports are added by the tutor. 8. The system displays the

		progress report for that week.
Alternate Flow	A6. If no weekly reports are available, the system displays a 'No reports available' message.	

UC-ID#	22	
Use Case Name	View Past Feedback for Tutor	
Primary Actor	Student, Parent	
Pre-condition	Feedback for the tutor is available in the system.	
Post-condition	The student, parent can view past feedback for the tutor.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. Student/Parent clicks on tutor profile card in home screen. 3. The Student/Parent clicks on reviews tab. 	<p>System Response:</p> <ol style="list-style-type: none"> 2. The system display tutors' profile with multiple tabs i.e. About, Packages and Reviews. 4. System retrieves and displays feedback which includes reviewer's name, rating (stars), review comments, and date/time of submission, average rating and total number of reviews.
Alternate Flow	A2. If no feedback is available, the system displays a 'No feedback available' message.	

UC-ID#	23	
Use Case Name	Provide Feedback for Tutor	
Primary Actor	Student, Parent	
Pre-condition	The specified number of weeks for the booked tutoring package have completed.	
Post-condition	Feedback for the tutor is saved in the database.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. Student/Parent clicks on bookings in navigation bar. 3. Student/Parent clicks on leave a review button on a booking in completed bookings section. 5. Student/Parent selects a rating and writes comments in the popup and clicks send feedback button. 	<p>System Response:</p> <ol style="list-style-type: none"> 2. System displays booking sections i.e. pending, active, completed and declined. 4. System displays a feedback popup which requires both a rating and comments. 6. The feedback is successfully stored in database.
Alternate Flow	<p>A6.1 If required fields (rating/comments) are empty, system displays an error message to fill in all fields.</p> <p>A6.2 If feedback submission fails, system displays an error and prompts to retry.</p>	

UC-ID#	24	
Use Case Name	Add Packages	
Primary Actor	Tutor	
Pre-condition	The tutor is logged into the app and navigates to earning section.	
Post-condition	A new package is successfully added and is available for students to book.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. The tutor clicks on earnings in navigation bar. 3. The tutor clicks on set packages button in my earnings screen. 5. The tutor clicks on add new package button in set packages screen. 7. The tutor enters Package Name, Package Description, Duration of Session, Sessions Per Week, Number of Weeks and Price. 8. The tutor clicks the save button. 	<p>System Response:</p> <ol style="list-style-type: none"> 2. The system displays my earnings screen. 4. The system display the set packages screen. 6. The system displays a form for the tutor to fill in the package details. 9. The system stores the package details to database.
Alternate Flow	A9.1 If any of the required fields are empty, the system prompts user to fill in all the fields.	

	<p>A9.2 If the system fails to store package details, system displays an error message.</p> <p>A9.3 If package description exceeds 150 characters, system displays an error message “Package description cannot exceed 150 characters”.</p> <p>A9.4 If hours and minutes are negative integers, the system displays an error message that “Hours and minutes must be non-negative integers”.</p> <p>A9.5 If minutes are less than 0 and greater than 59, system displays an error message showing “Minutes should be between 0 and 59”.</p> <p>A9.6 If sessions per week is less than equal to zero, system displays an error message showing “Sessions per week must be greater than zero”.</p> <p>A9.7 If number of weeks is less than equal to zero, system displays an error message showing “Number of weeks must be greater than zero”.</p> <p>A9.8 If price is less than equal to zero, system displays an error message showing “Price must be greater than zero”.</p>
--	---

UC-ID#	25	
Use Case Name	Report User	
Primary Actor	Student, Parent, Tutor	
Pre-condition	There is already an ongoing conversation with the user to be reported.	
Post-condition	The user receives confirmation about report submission and report is sent to admin for review.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> The user navigates to chat list screen where all chats are displayed. 	<p>System Response:</p> <ol style="list-style-type: none"> The system displays list of chats.

	<p>3. User long taps on a specific chat to report the user.</p> <p>5. User clicks report option.</p> <p>7. The user fills in the comments field and attaches images.</p> <p>8. The user clicks on report user button.</p>	<p>4. System displays a bottom sheet with the option to report.</p> <p>6. The system displays a popup menu with comments field and an option to attach images.</p> <p>9. The system successfully stores the report in database and sends it to admin for review.</p>
Alternate Flow	<p>A9.1 If the system fails to store and submit the report for review, the system displays an error message.</p> <p>A9.2 If the user doesn't fill the comments field, system prompts the user to "Fill in the comments field".</p> <p>A9.3 If the user doesn't select and add images, system prompts the user "Please attach images"</p> <p>A9.4 If the user selects a file type other than png and jpg, system prompts the user to "Select a valid file type".</p>	

UC-ID#	26	
Use Case Name	Block User	
Primary Actor	Admin	
Pre-condition	The admin is logged in to the application.	
Post-condition	The targeted user is blocked preventing them from logging in to the mobile app.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. The admin clicks on manage users in side navigation bar. 3. The admin turns on the toggle button for the specific user to be blocked. 	<p>System Response:</p> <ol style="list-style-type: none"> 2. The system displays all list of users available on app including their name, user id, email, user type and block user toggle button and chat icon. 4. The system successfully blocks the user showing “User blocked successfully”
Alternate Flow	A4. If the system fails to block the user, the system displays an error message “Failed to block user try again”	

UC-ID#	27	
Use Case Name	Unblock User	
Primary Actor	Admin	
Pre-condition	The admin is logged in to the application.	
Post-condition	The targeted user is unblocked allowing them to log in to the mobile app.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. The admin clicks on manage users in the side navigation bar. 	<p>System Response:</p>

	<p>3. The admin turns off the toggle button for the specific user to be unblocked.</p>	<p>2. The system displays all list of users available on app including their name, user id, email, user type and block user toggle button and chat icon.</p> <p>4. The system successfully unblocks the user showing "User unblocked successfully"</p>
Alternate Flow	A4. If the system fails to unblock the user, the system displays an error message.	

UC-ID#	28	
Use Case Name	Archive chat	
Primary Actor	Tutor, Student, Parent, Admin	
Pre-condition	There is already an ongoing conversation with the user to archive that chat.	
Post-condition	The selected chat is moved to the "Archived Chats" section and removed from the active chats list.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. The user navigates to chat list screen by clicking chat icon. 3. The user long taps on the chat to be archived. 4. The user clicks on archive chat option. 	<p>System Response:</p> <ol style="list-style-type: none"> 2. The system displays list of chats. 4. The system displays a bottom sheet with an option to archive chat.

		5. The system successfully archives the chat and moves the chat to archive chat section and removes it from chat list.
Alternate Flow	A5. If the system fails to archive a chat, the system displays an error message showing "Failed to archive chat. Try again."	

UC-ID#	29	
Use Case Name	Unarchive chat	
Primary Actor	Student, Parent, Tutor, Admin	
Pre-condition	The chat already exists in archive chats section.	
Post-condition	The selected chat is moved to active list of chats and removed from the archived chats section.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. The user navigates to chat list screen by clicking chat icon. 3. The user navigates to archive chats section by clicking the archive chat icon. 5. The user long taps on the chat to be unarchived. 4. The user clicks on unarchive chat option. 	<p>System Response:</p> <ol style="list-style-type: none"> 2. The system displays list of chats. 4. The system displays archived list of chats. 6. The system displays a bottom sheet with an option to unarchive chat. 5. The system successfully unarchives the chat and moves the chat to active list

		of chats and removes it from archived chats section.
Alternate Flow	A5. If the system fails to unarchive a chat, the system displays an error message showing "Failed to unarchive chat. Try again."	

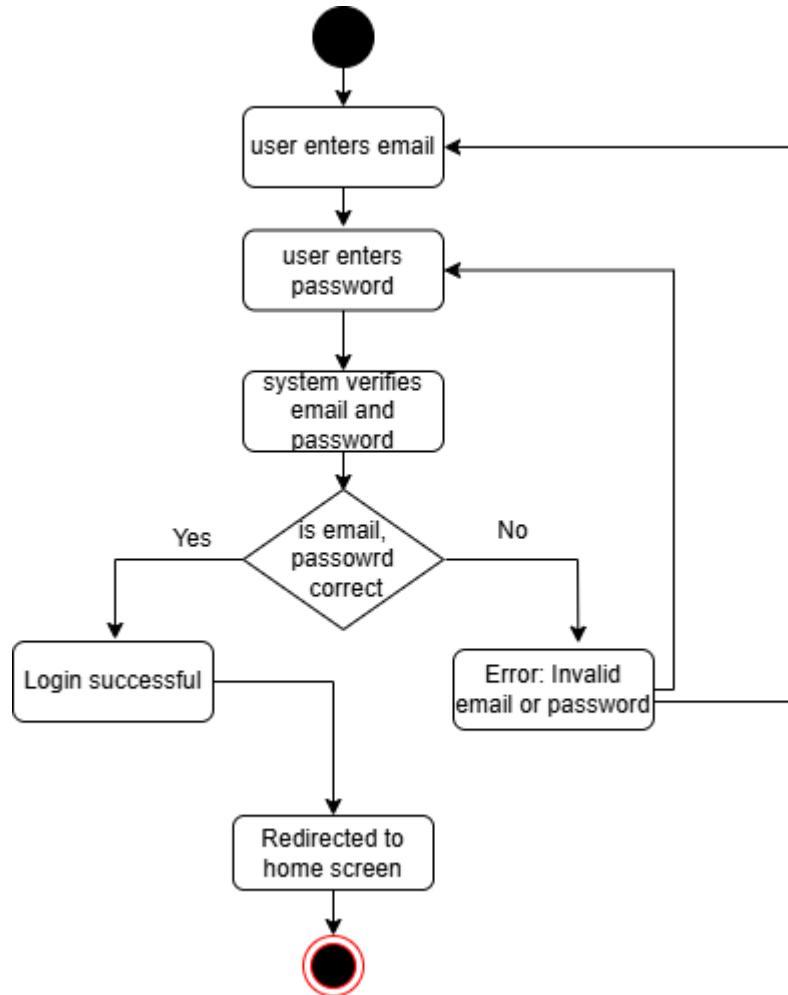
UC-ID#	30	
Use Case Name	Pin chat	
Primary Actor	Tutor, Student, Parent, Admin	
Pre-condition	There is already an ongoing conversation with the user to pin that chat.	
Post-condition	The selected chat is pinned and displayed at the top of the chat list, marked with a pin icon.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. The user navigates to chat list screen by clicking chat icon. 3. The user long taps on the chat to be pinned. 4. The user clicks on pin chat option. 	<p>System Response:</p> <ol style="list-style-type: none"> 2. The system displays list of chats. 4. The system displays a bottom sheet with an option to pin chat. 5. The system successfully pins the chat and moves the chat above all other chats marked with a pin icon.
Alternate Flow	A5. If the system fails to pin a chat, the system displays an error message showing "Failed to pin chat. Try again."	

UC-ID#	31	
Use Case Name	Unpin chat	
Primary Actor	Student, Parent, Tutor, Admin	
Pre-condition	The chat to be unpinned is already displayed above all other chats marked with a pin icon.	
Post-condition	The selected chat is unpinned and moved back to its original position in the chat list chronologically, and the pin icon is removed.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. The user navigates to chat list screen by clicking chat icon. 3. The user long taps on the desired pinned chat to unpin. 5. The user clicks on unpin chat option. 	<p>System Response:</p> <ol style="list-style-type: none"> 2. The system displays list of chats. 4. The system displays a bottom sheet with an option to unpin chat. 6. The system successfully unpins the chat and moves it back to its original position in the chat list chronologically, and the pin icon is removed.
Alternate Flow	A6. If the system fails to unpin a chat, the system displays an error message showing "Failed to unpin chat. Try again."	

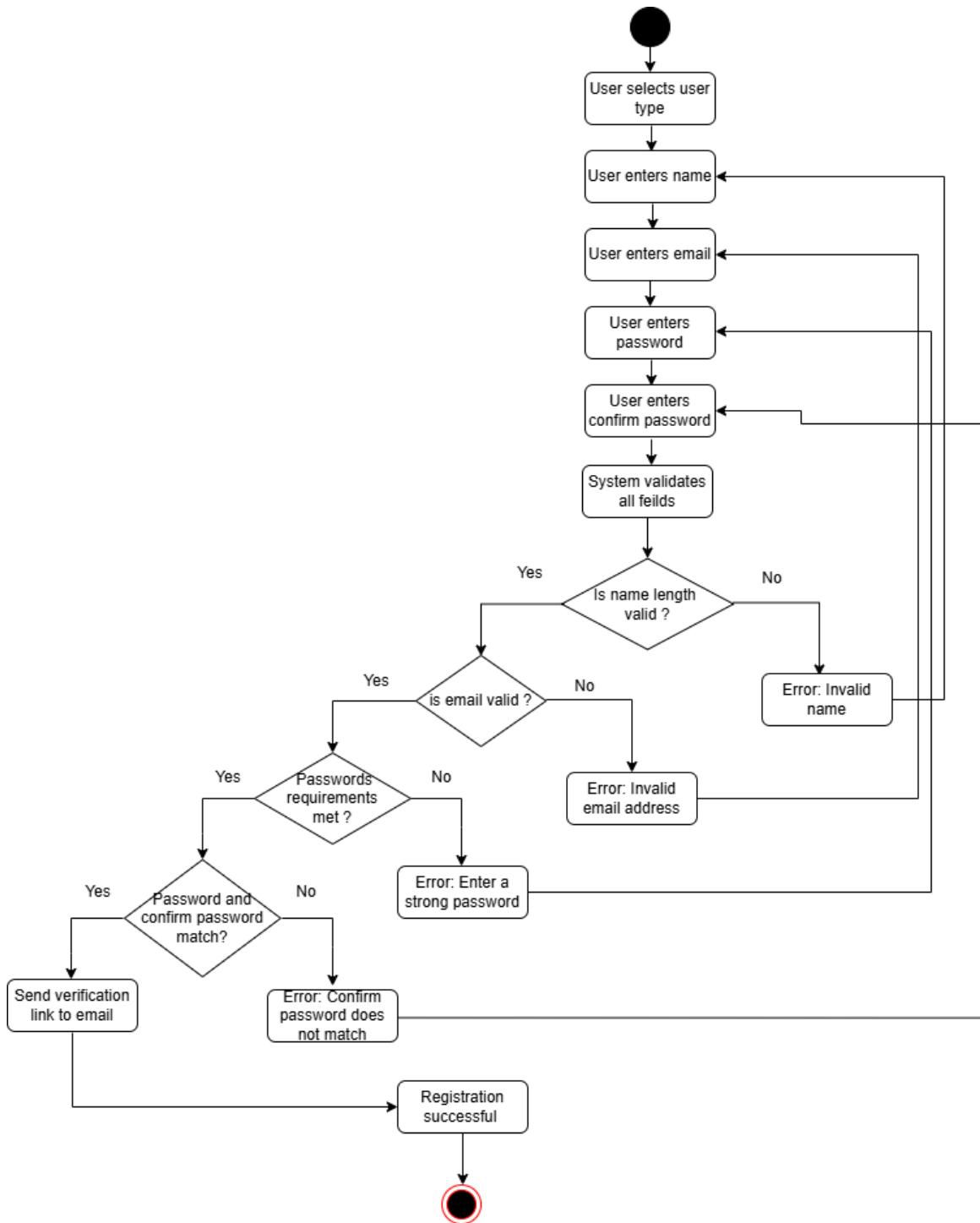
UC-ID#	32	
Use Case Name	Resolve reports	
Primary Actor	Admin	
Pre-condition	Admin must be logged in to the application.	
Post-condition	Suitable action is taken (blocked, warned, marked as spam, resolved) which updates report status to closed from open.	
Normal Flow	<p>User Action:</p> <ol style="list-style-type: none"> 1. The admin clicks on reports in the side navigation bar. 3. The admin clicks the view button for specific report which he wants to review. 5. The admin clicks on the suitable action to mark the report as closed: <ul style="list-style-type: none"> • Blocked (blocks the reported user) • Warned (sends a warning message to the reported user) • Marked as spam • Resolved 	<p>System Response:</p> <ol style="list-style-type: none"> 2. The system displays a table with a list of all the reports which have been reported. 4. The system displays all the details related to the report, report history of both reporter and reported by user and provide admin options to take on reports. 6. The system successfully takes the action on the reported user and changes its status to close.
Alternate Flow	A6. If the system fails to resolve a report, the system displays an error message showing "Failed to resolve report. Try again."	

4.4 Activity Diagrams

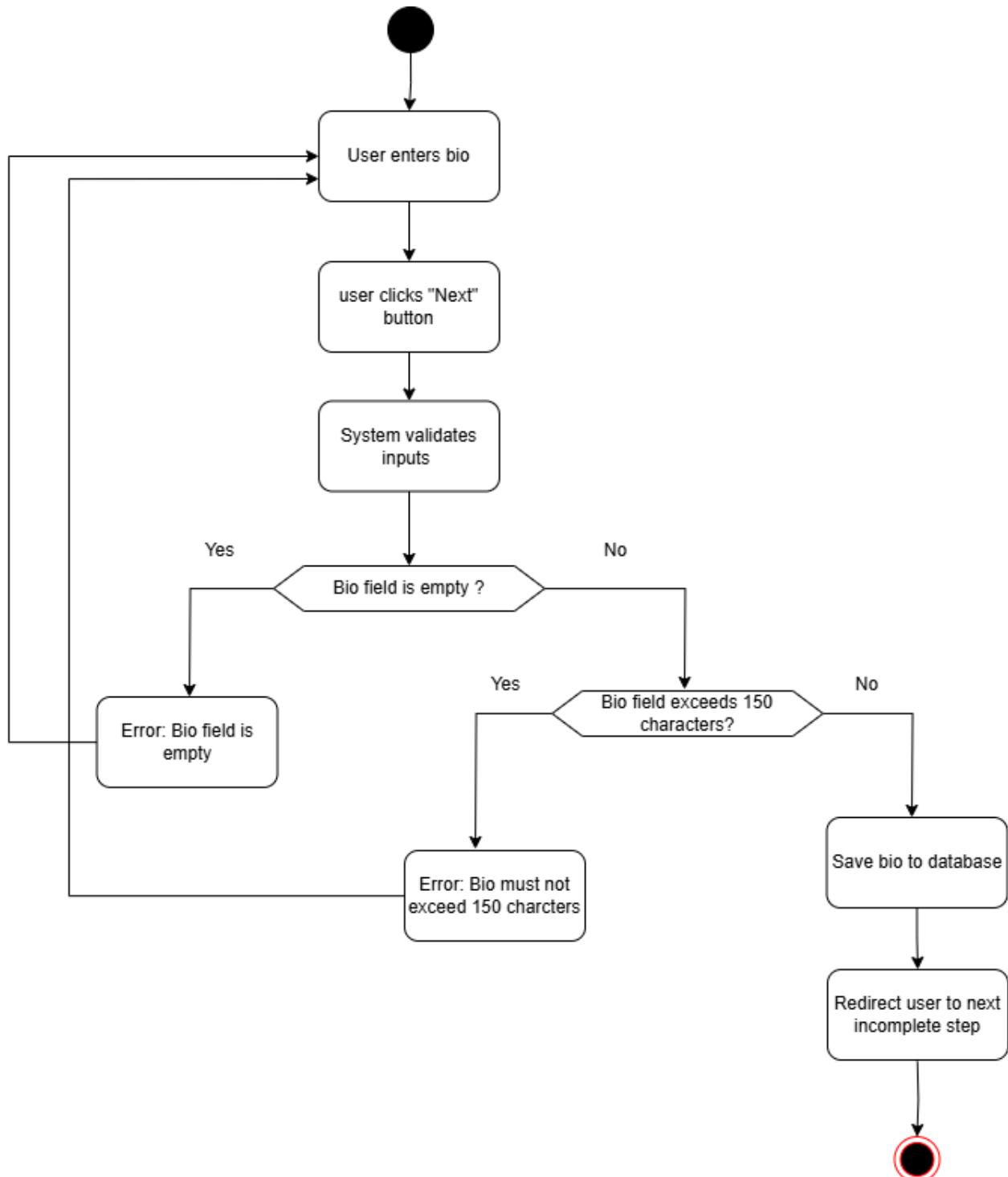
Use case 01: Login Account



Use case 02: Register Account



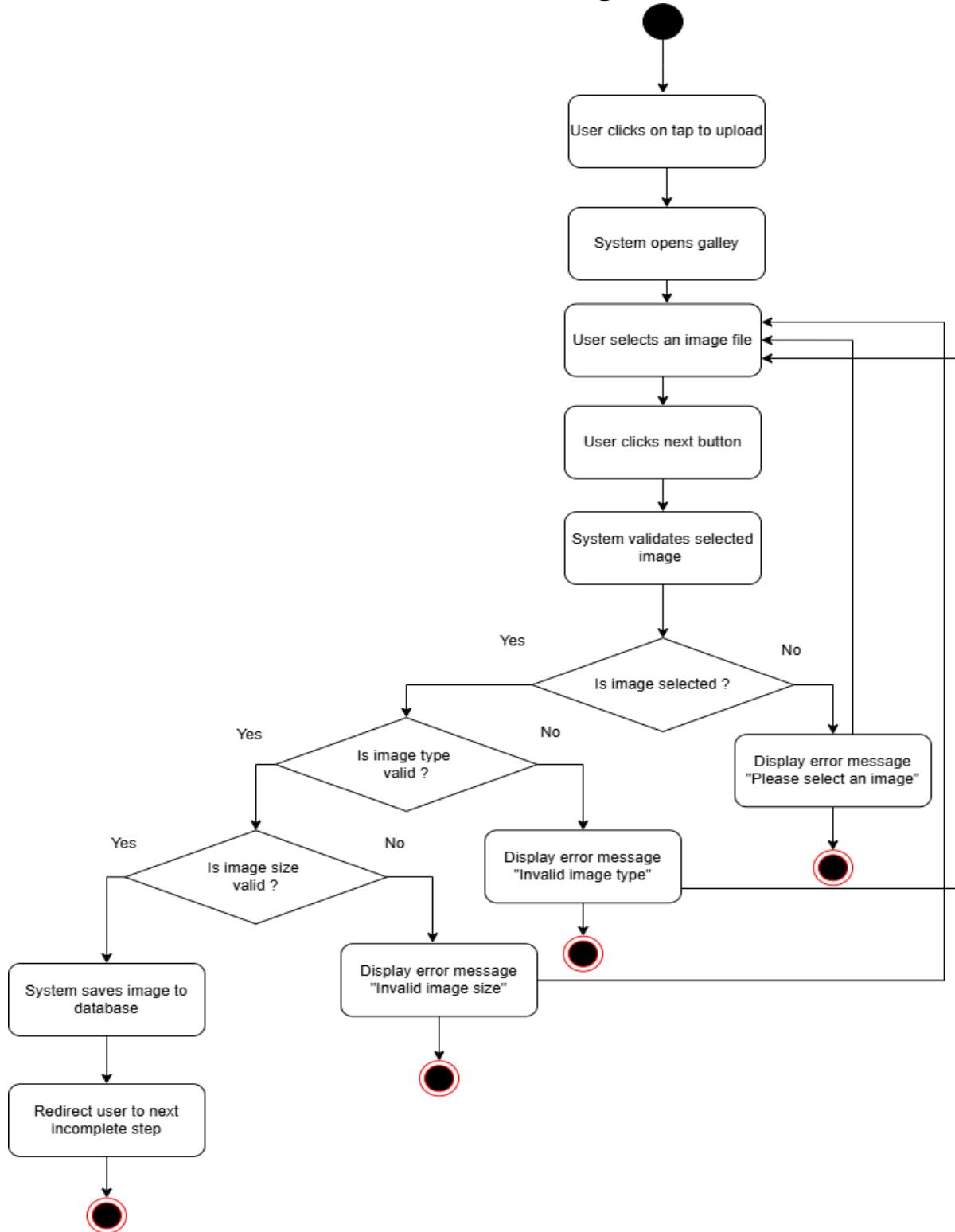
Use case 03: Add Bio



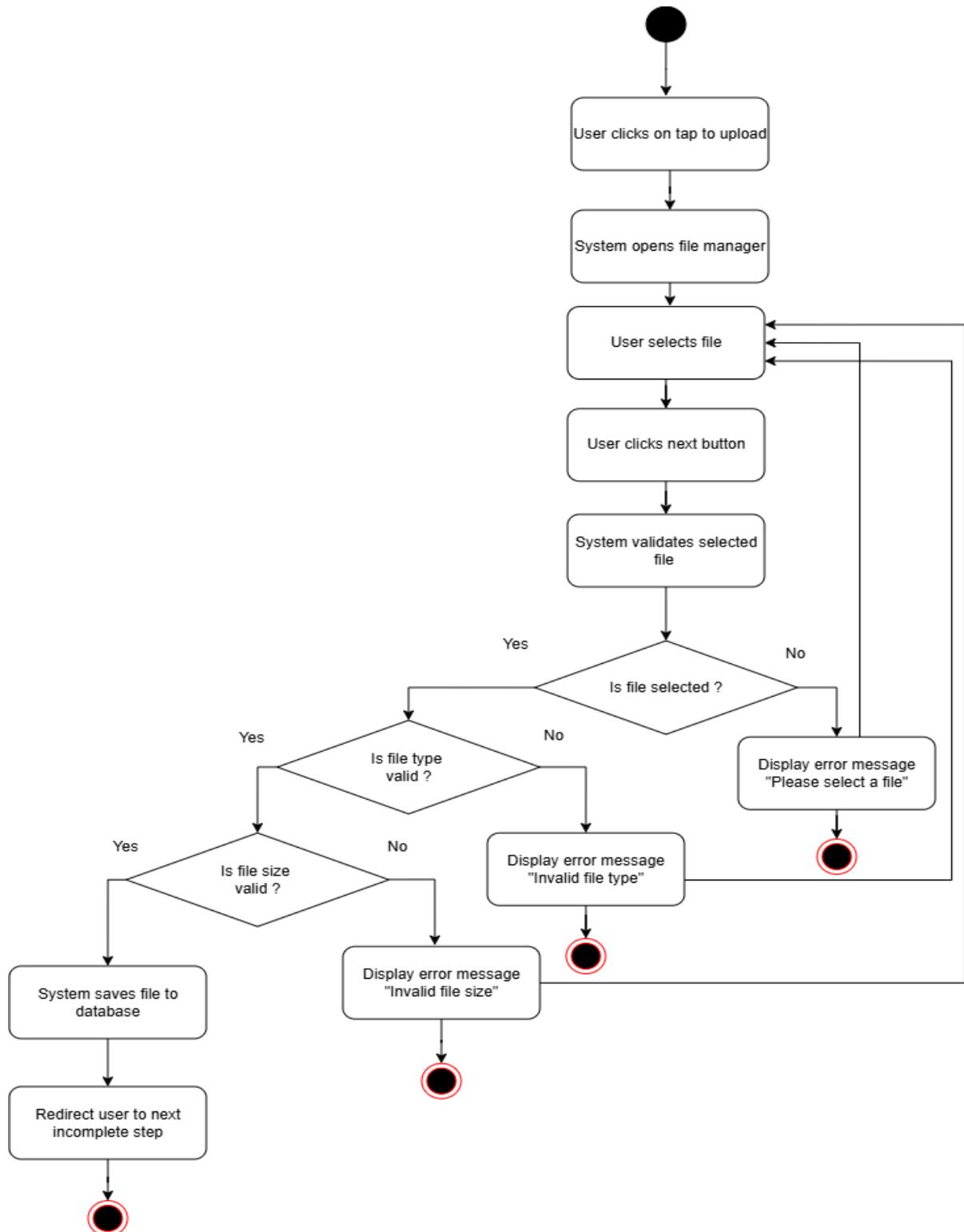
Use case 04: Add Location



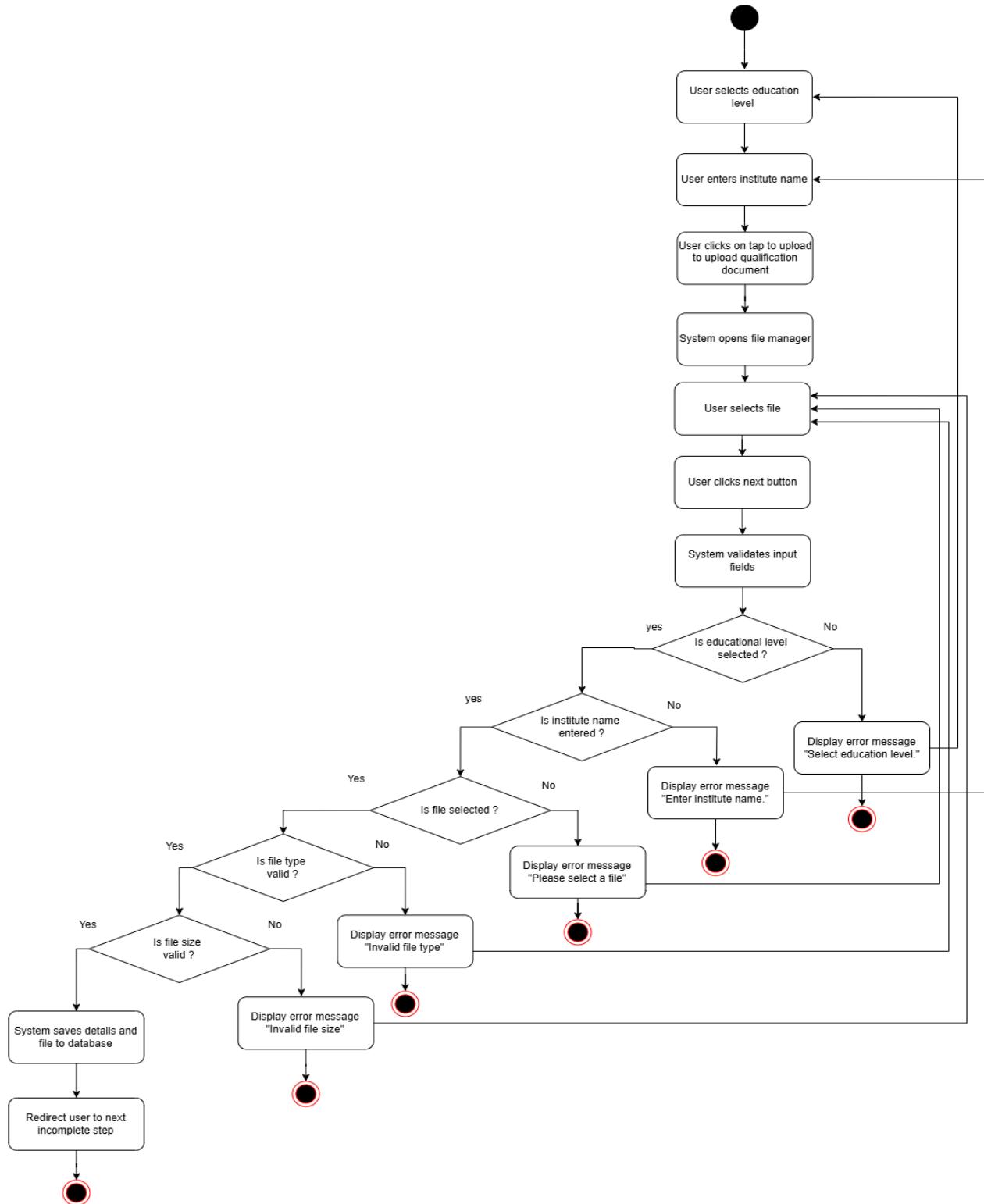
Use case 05: Add Image



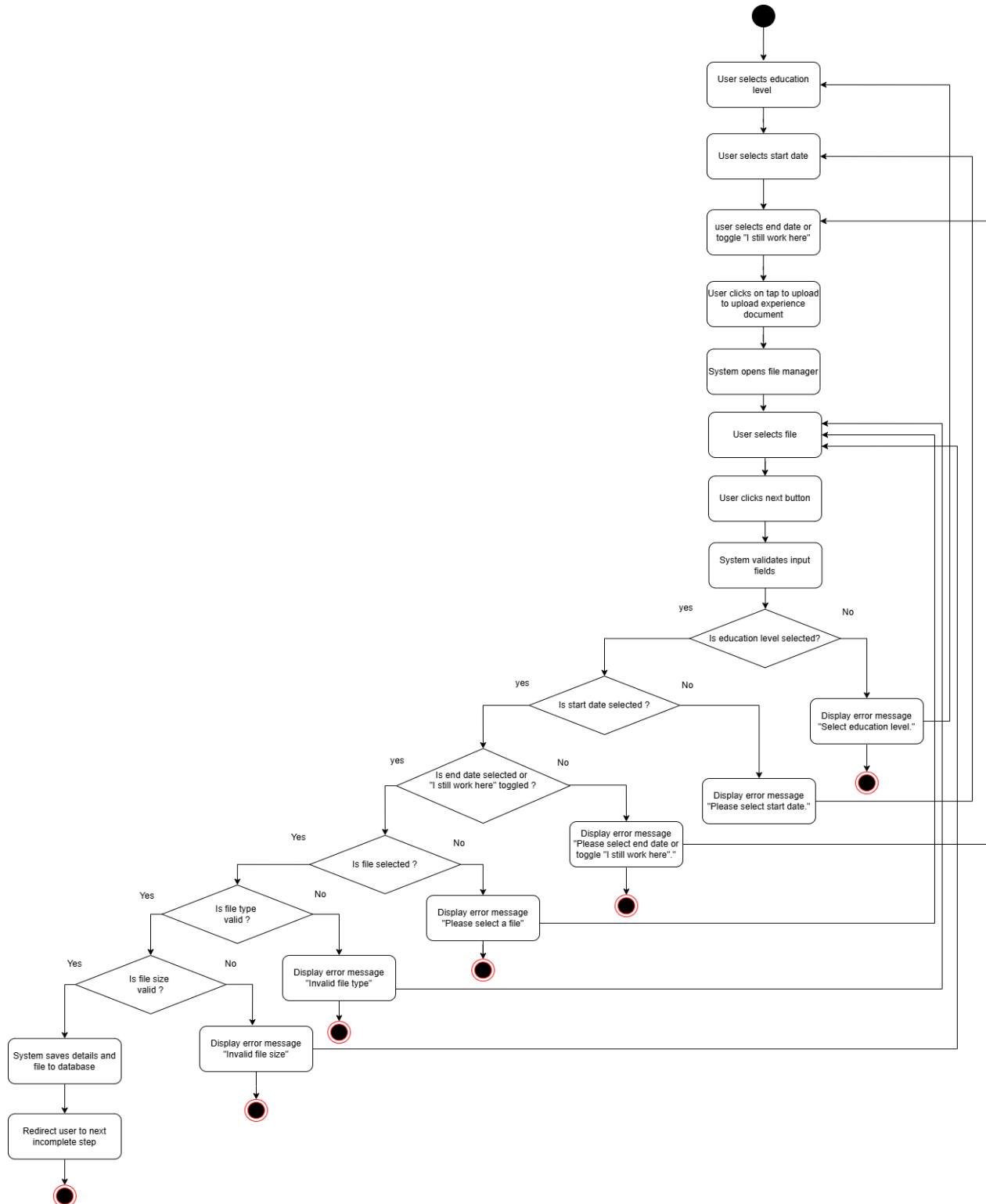
Use case 06: Add CNIC



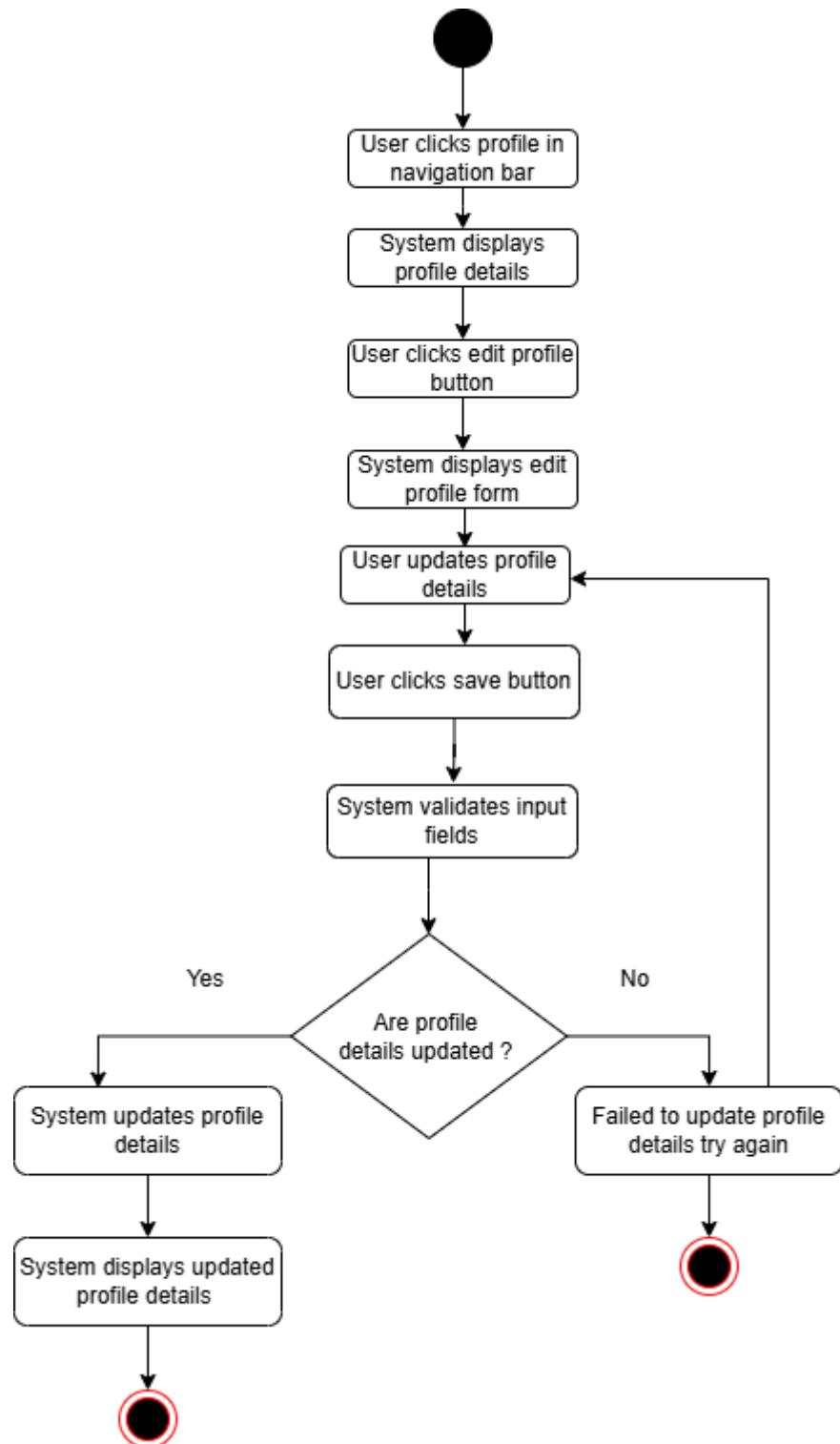
Use case 07: Add Qualification



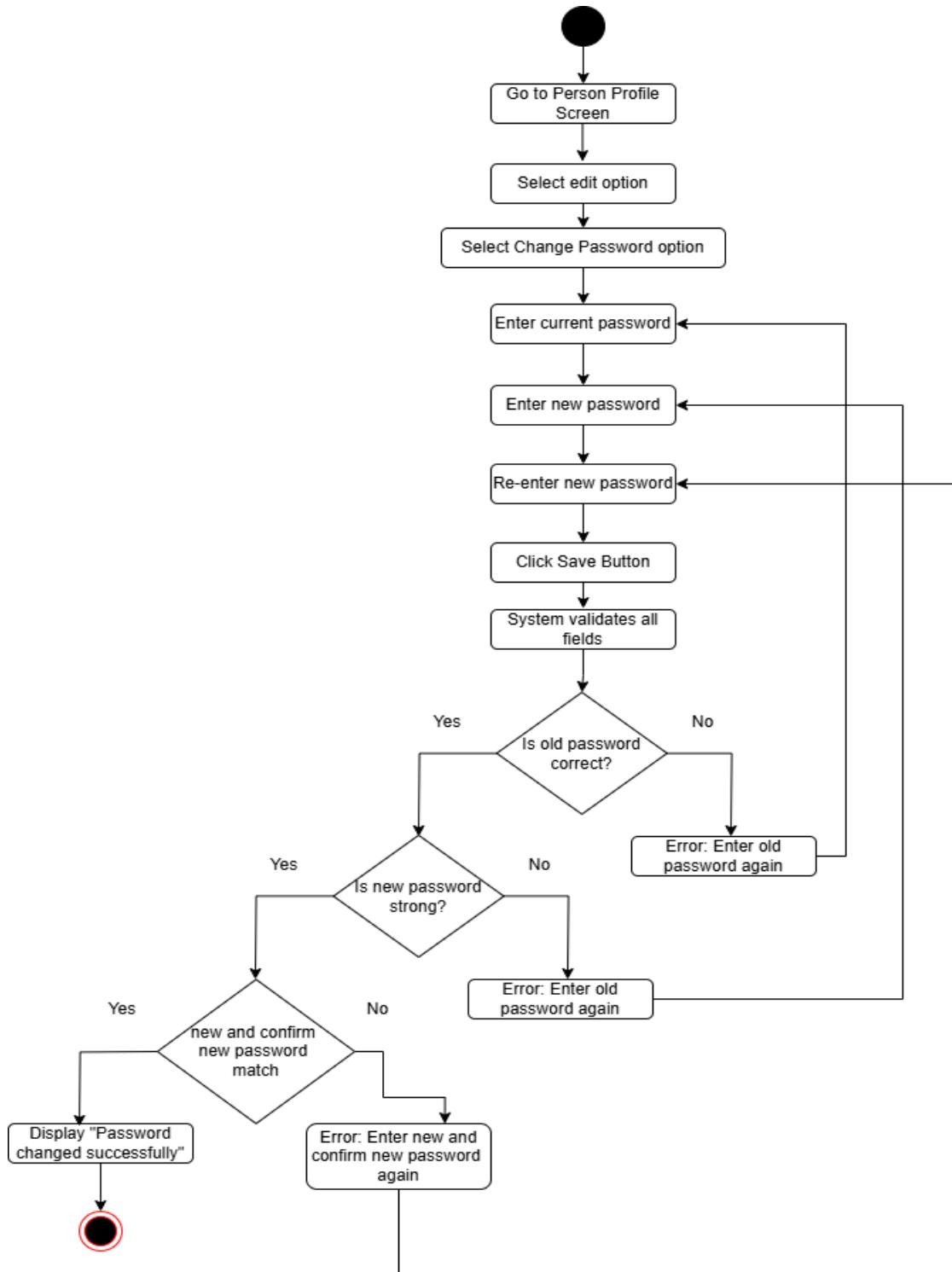
Use case 08: Add Experience



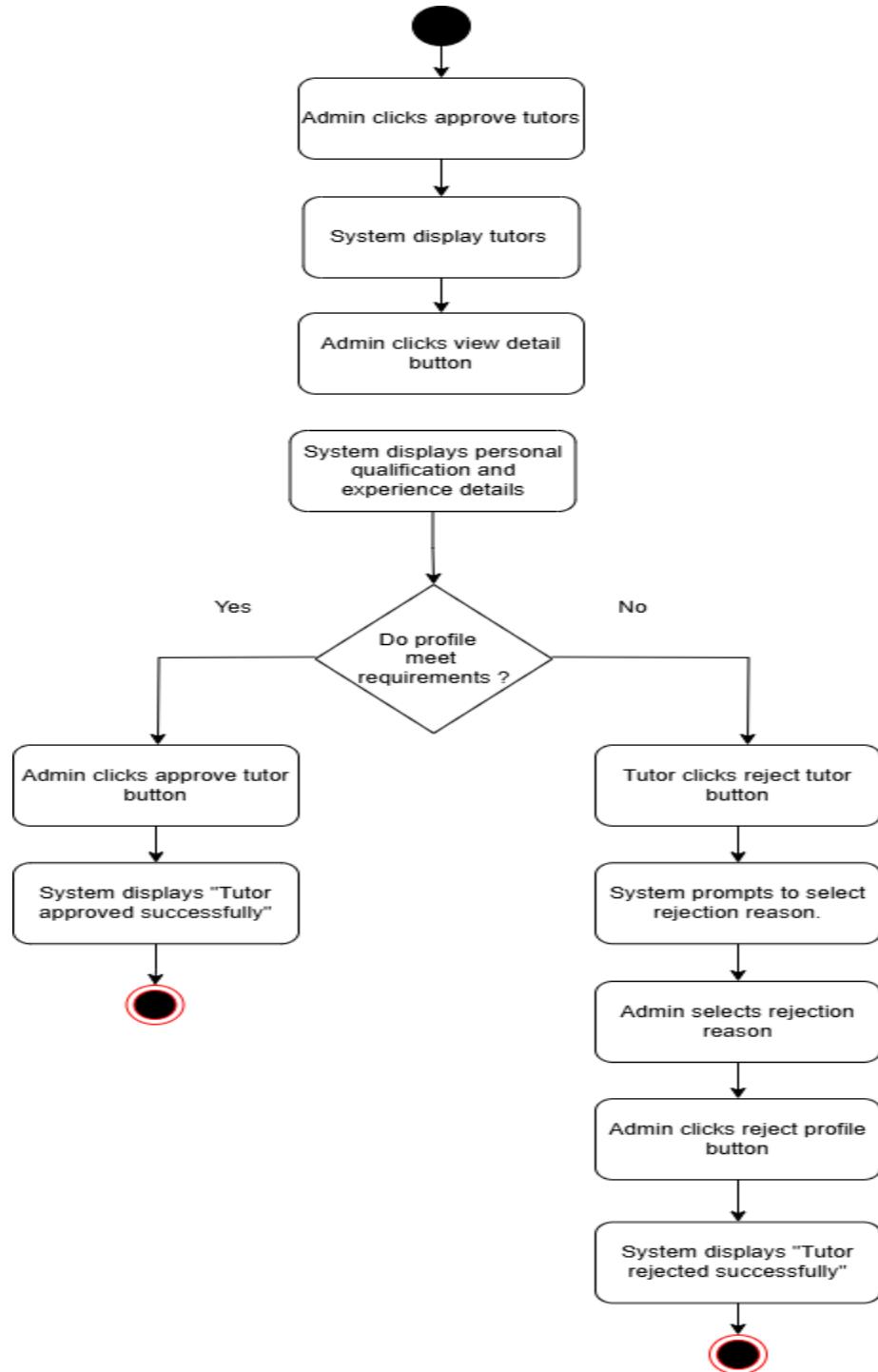
Use case 09: Edit Profile



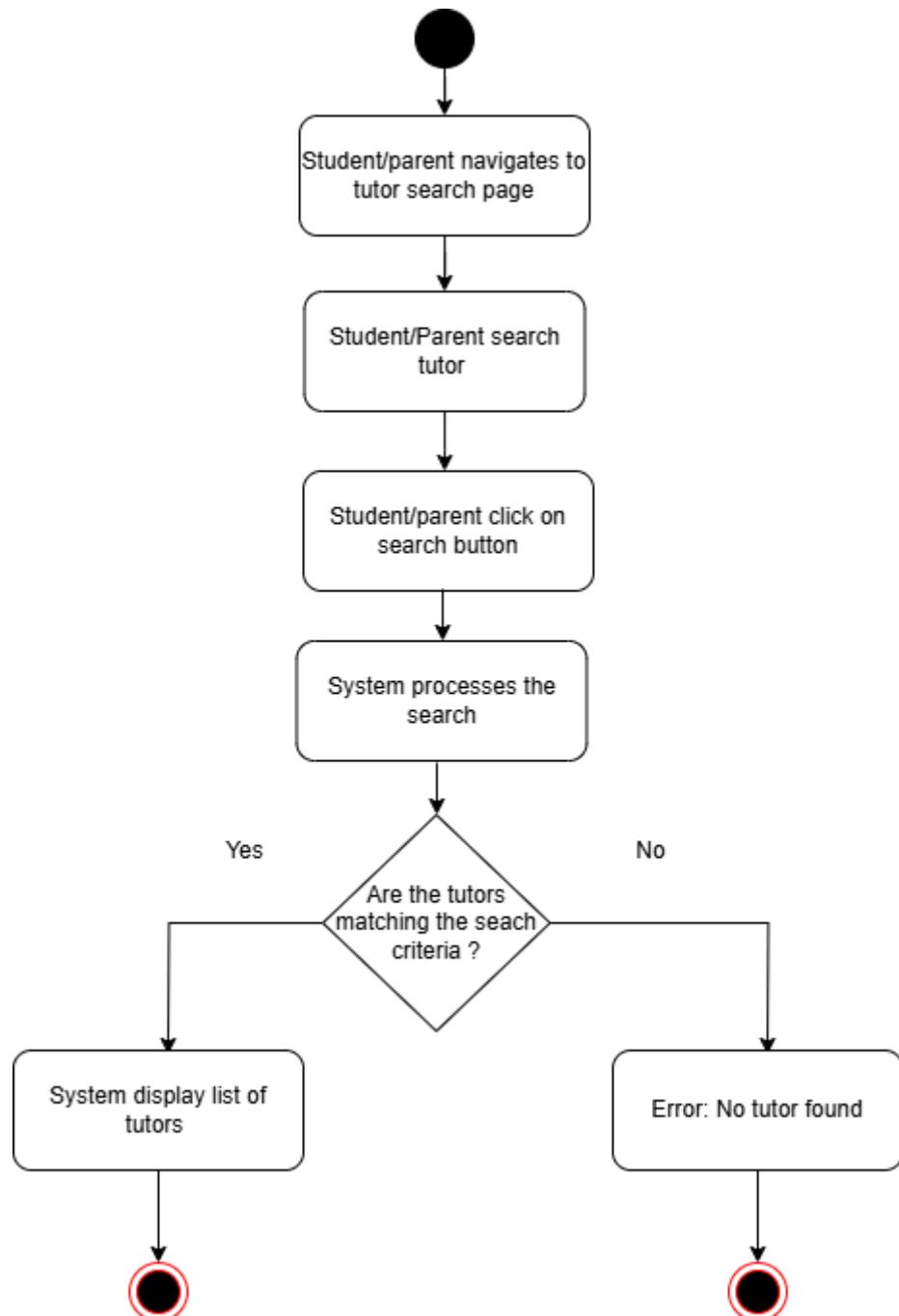
Use case 10: Change Password



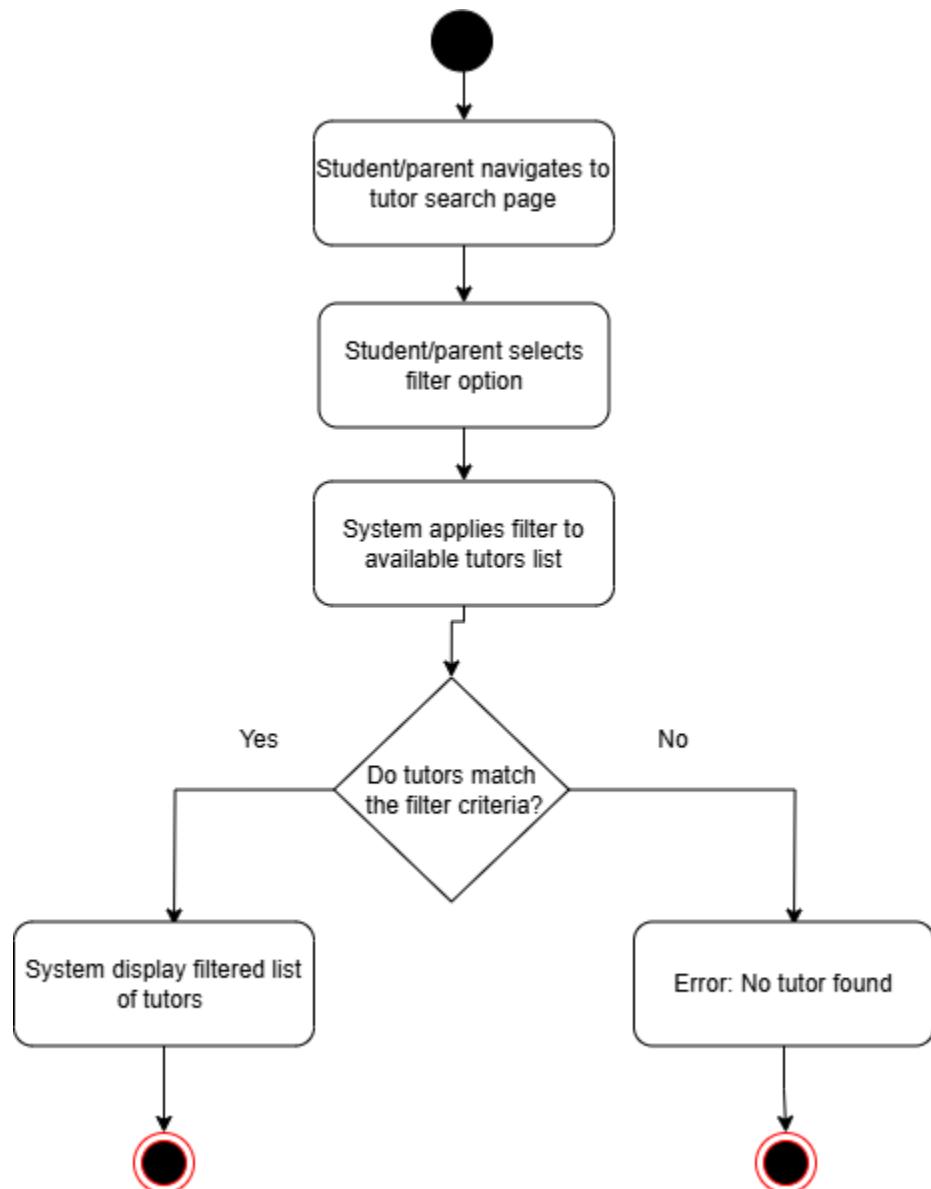
Use case 11: Verify tutor profile



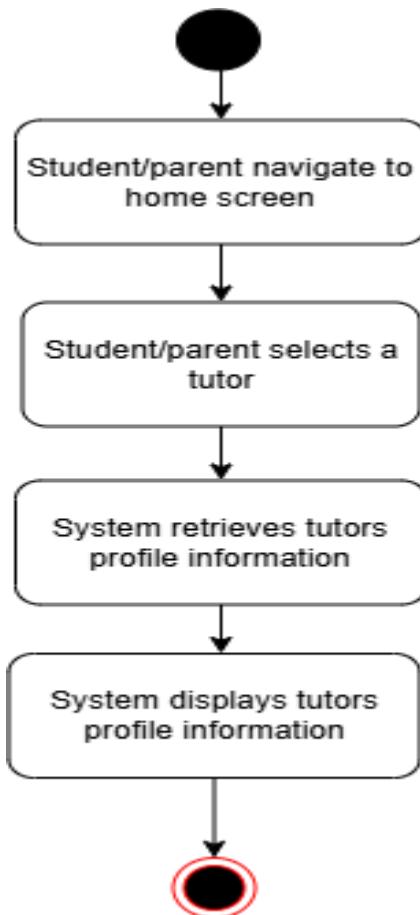
Use case 12: Search tutor



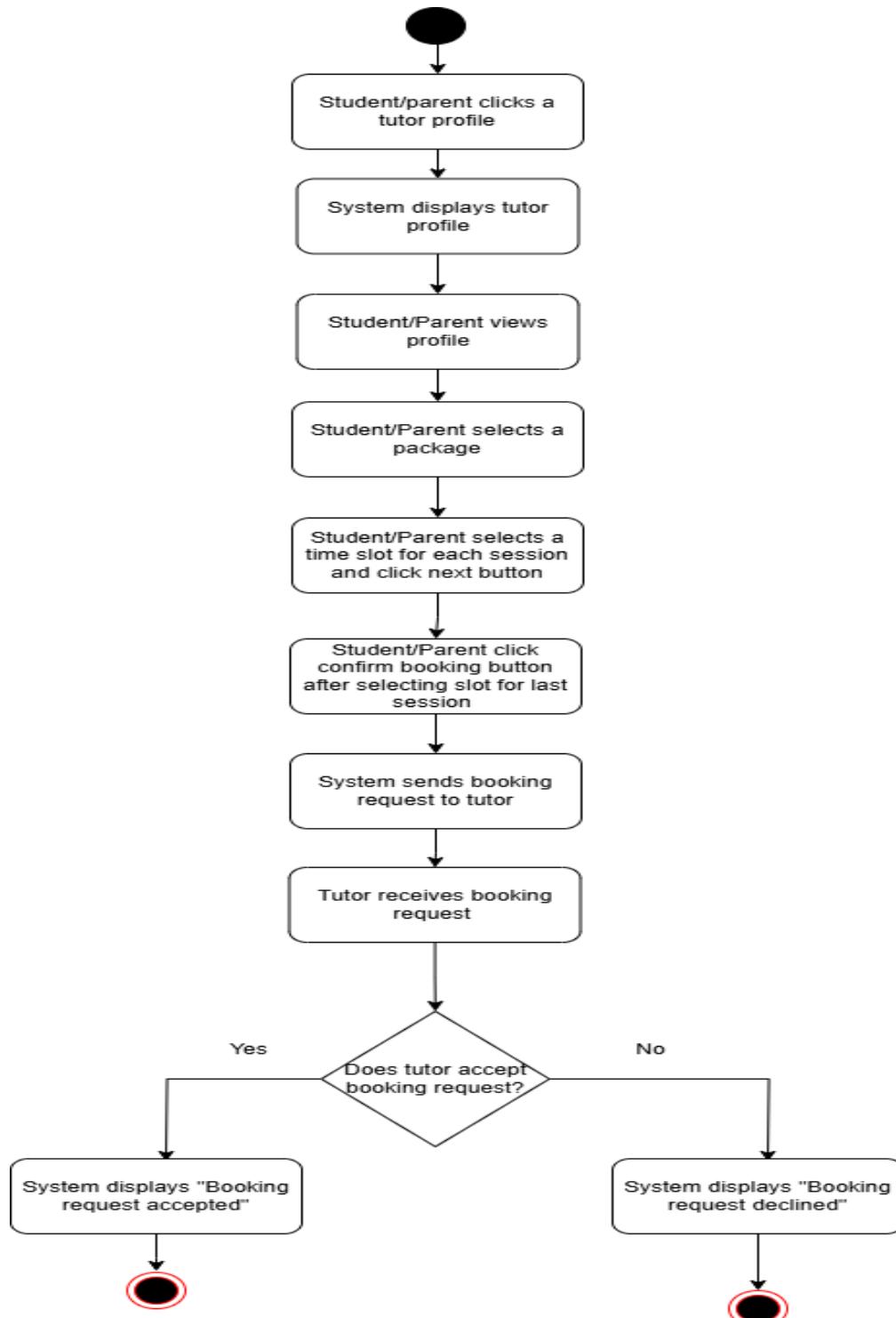
Use case 13: Filter tutor



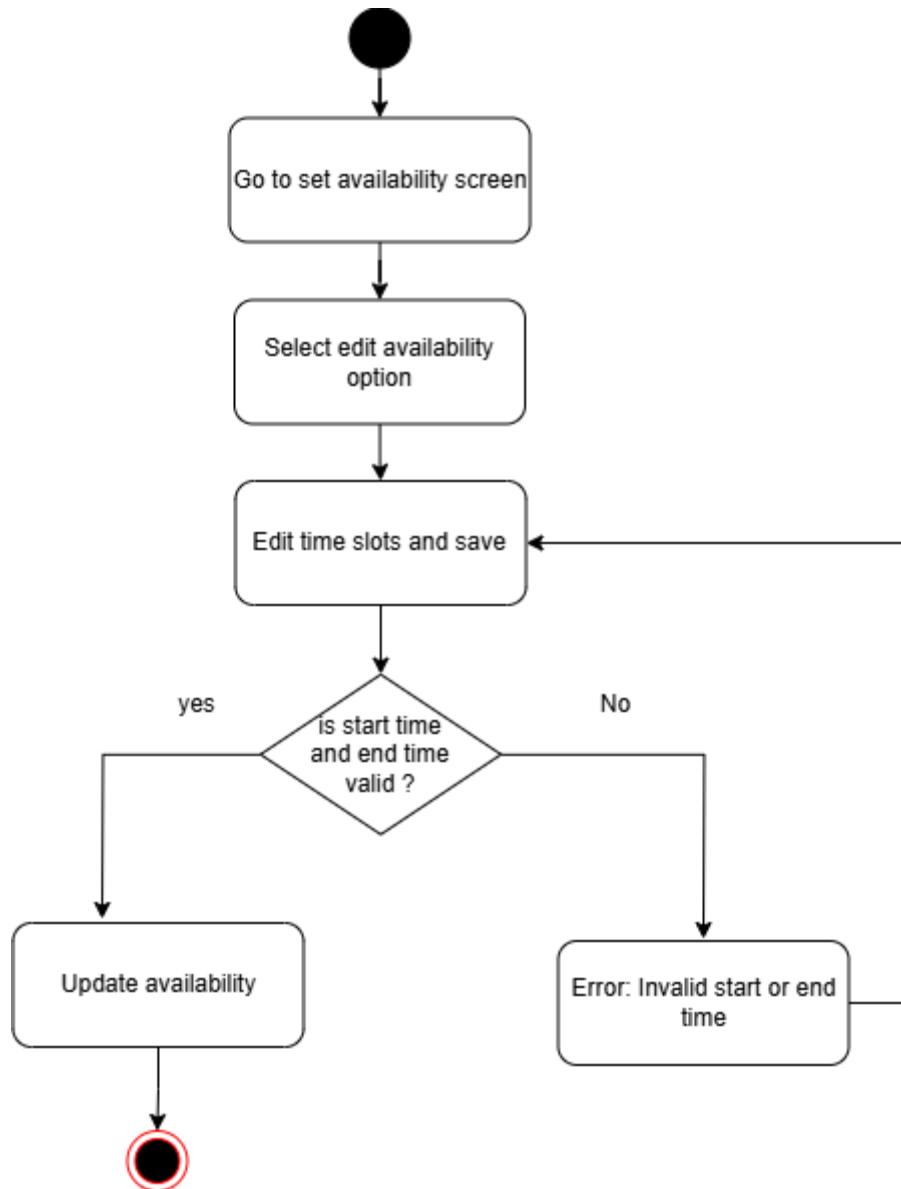
Use case 14: View tutor Profile



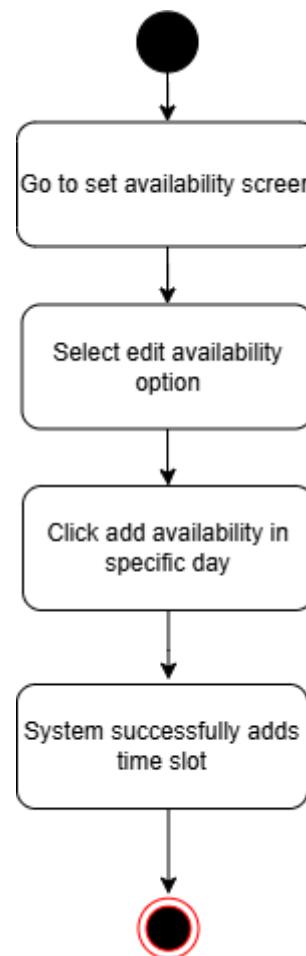
Use case 15: Book tutor



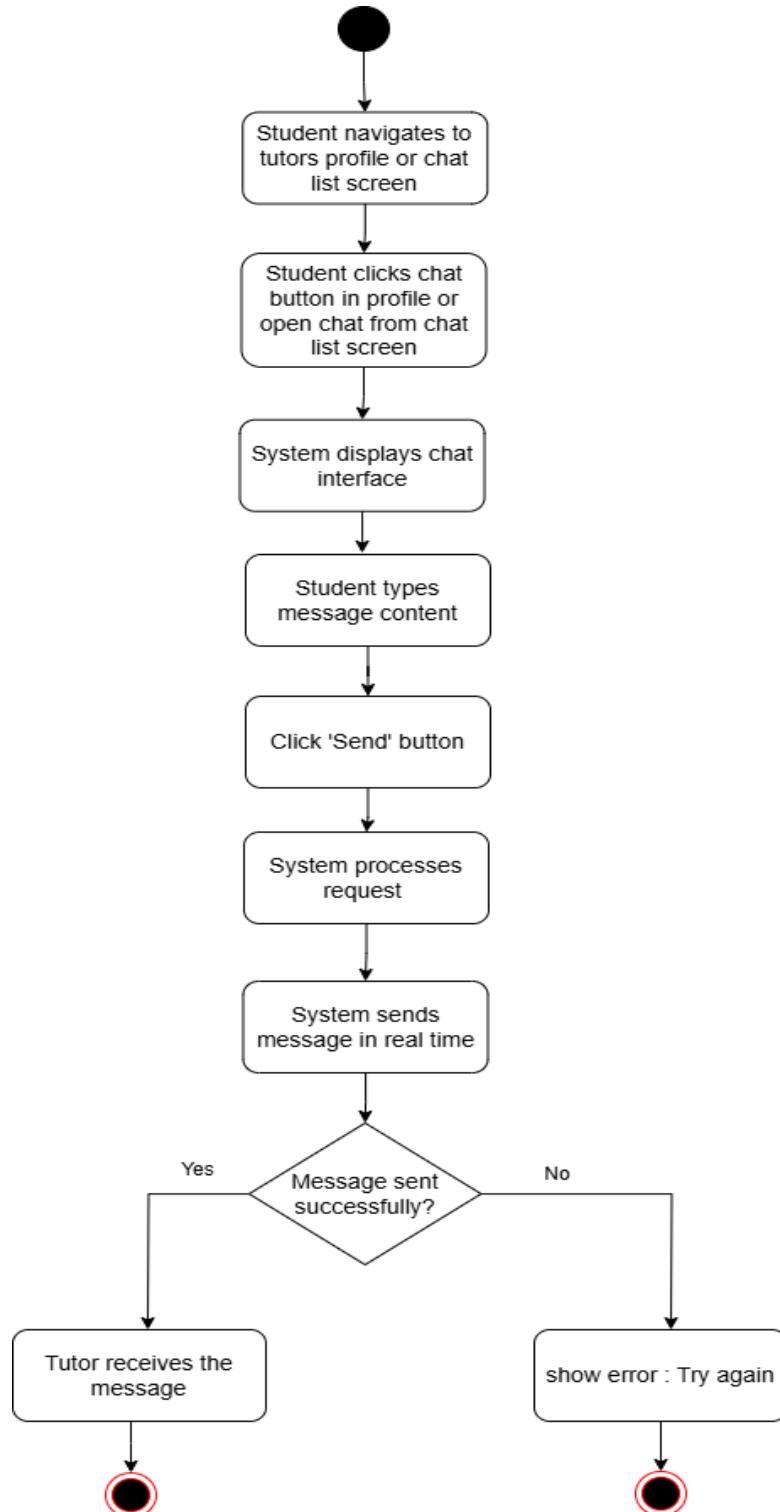
Use case 16: Edit Availability



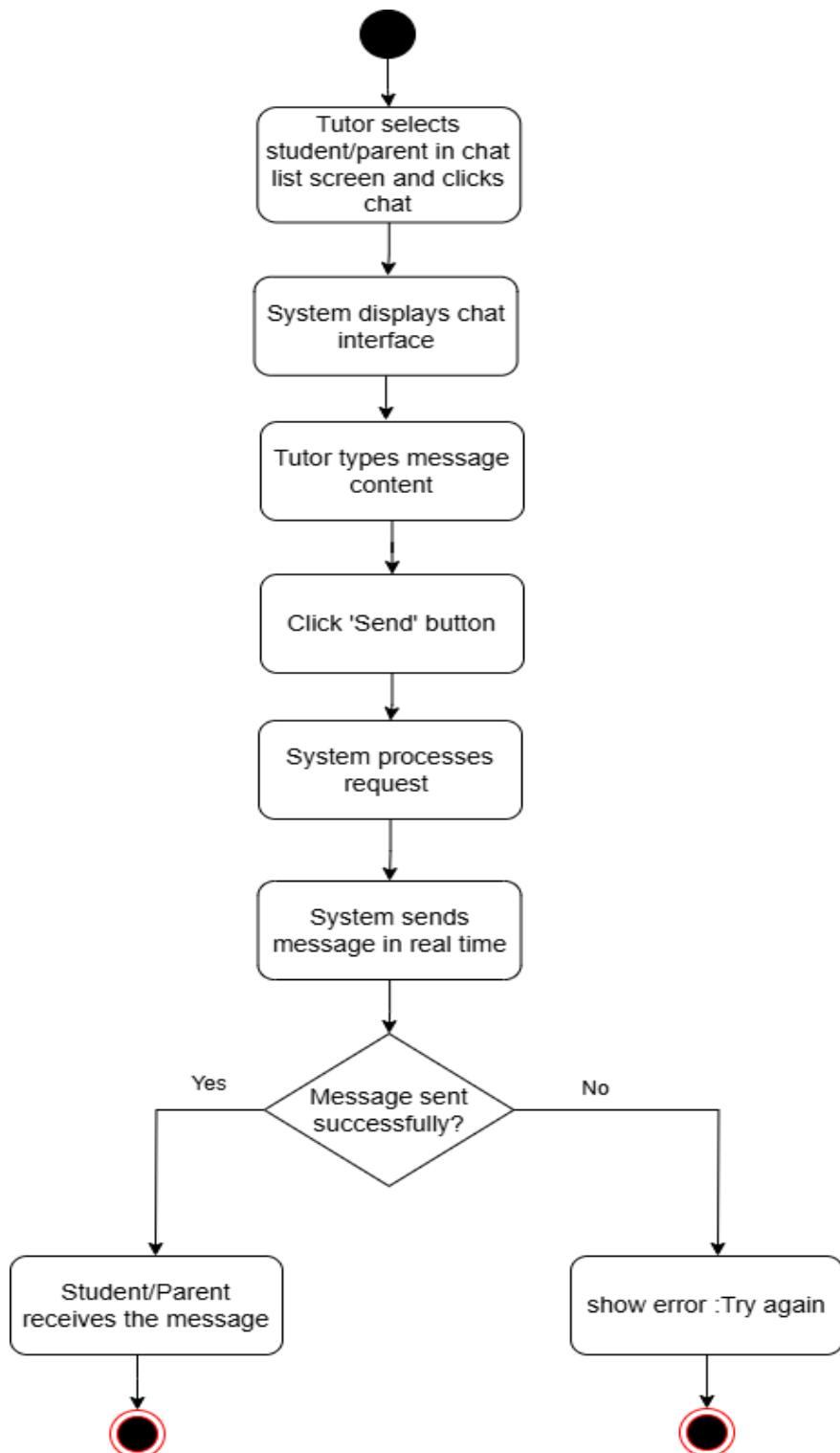
Use case 17: Add time slots



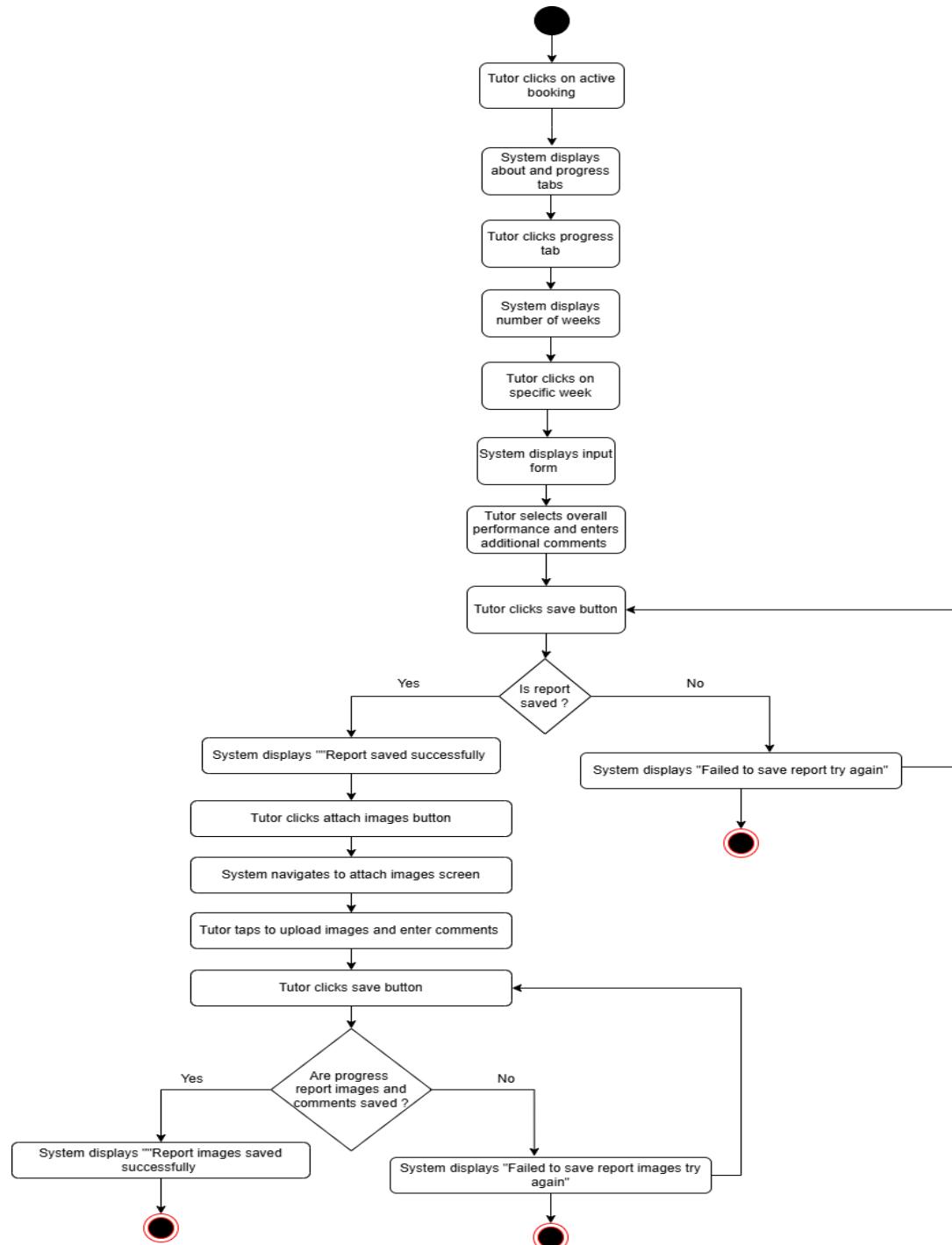
Use case 18: Send message to tutor



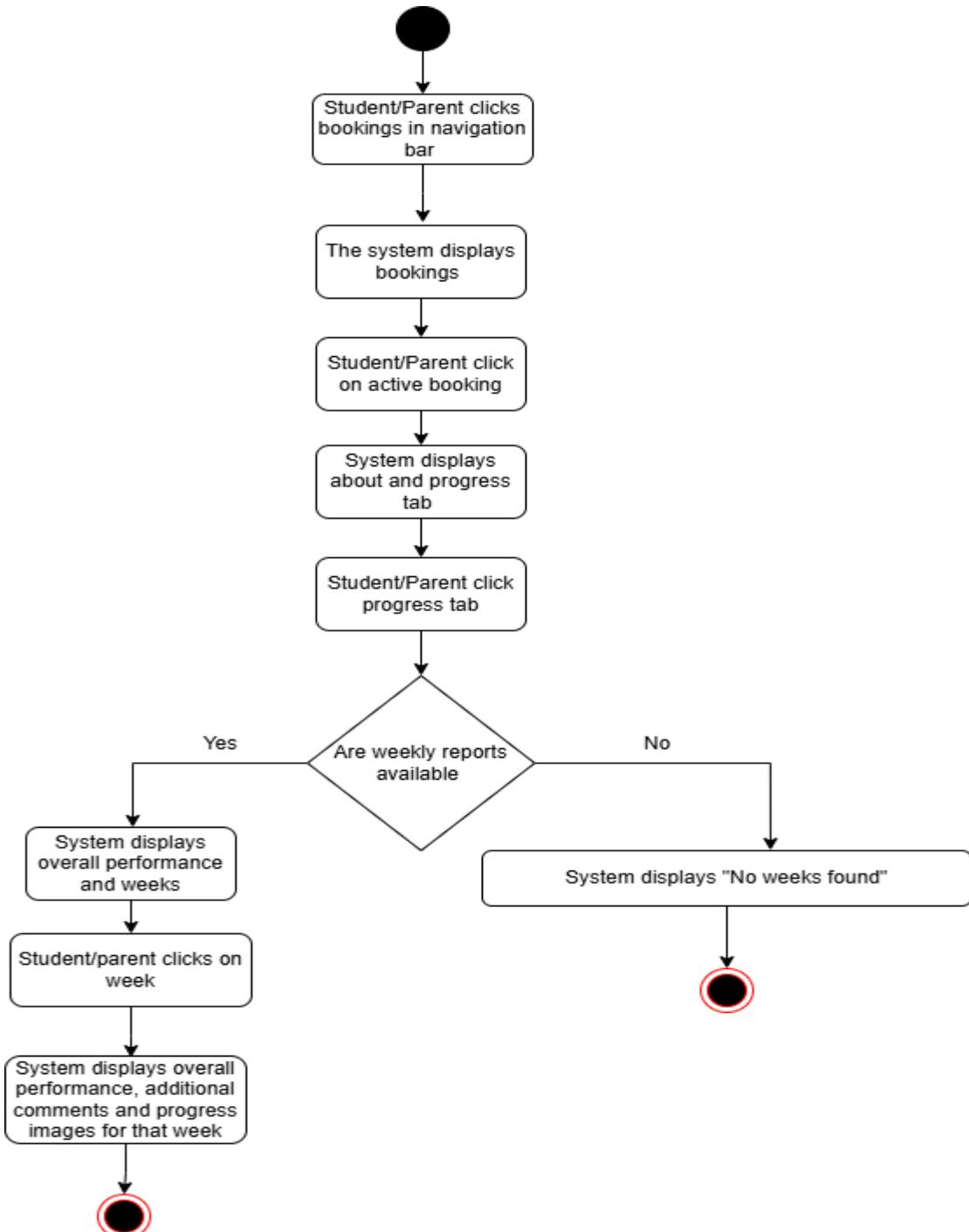
Use case 19: Send message to student/parent



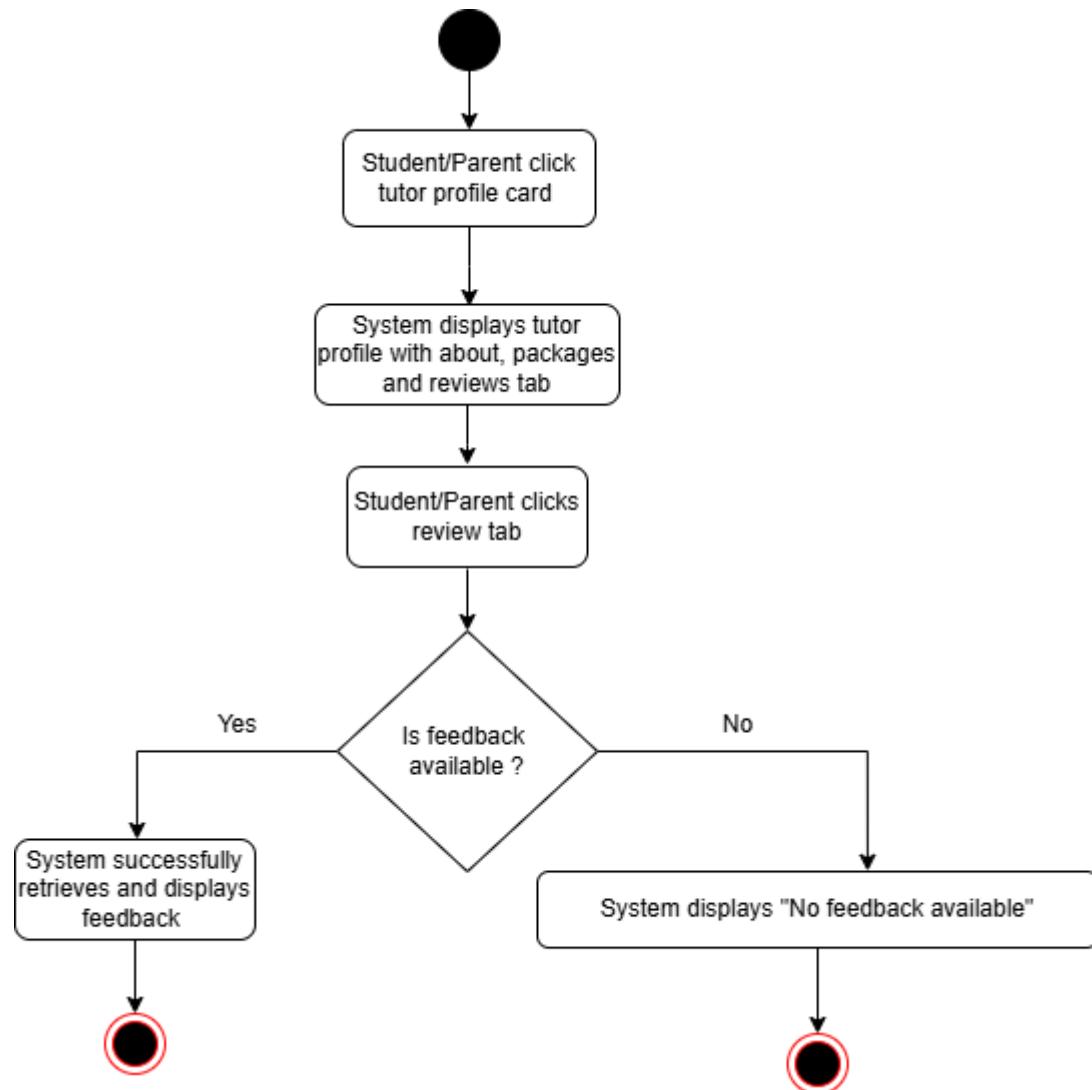
Use case 20: Generate Weekly Progress Reports



Use case 21: View Weekly progress reports



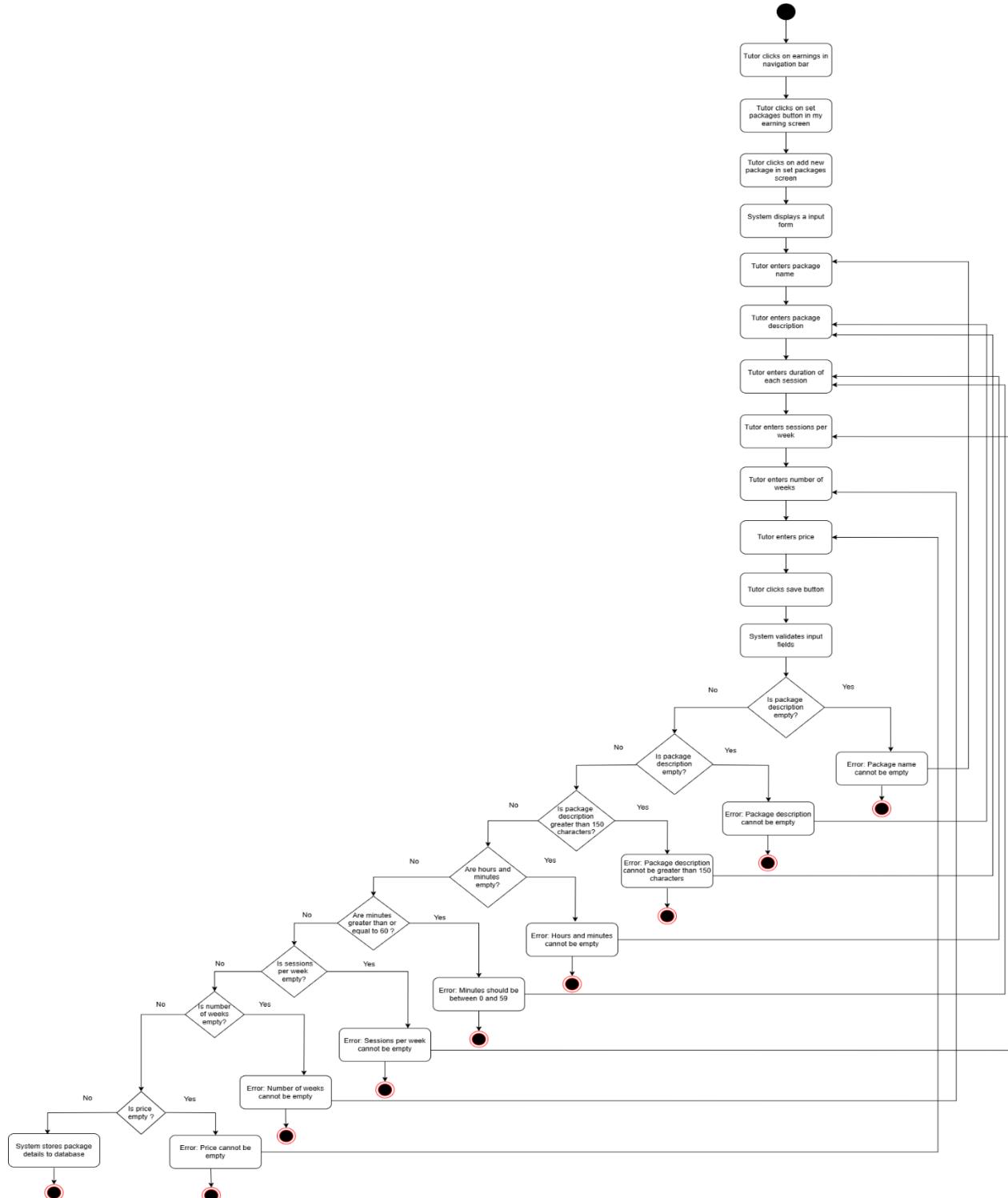
Use case 22: View past feedback for tutor



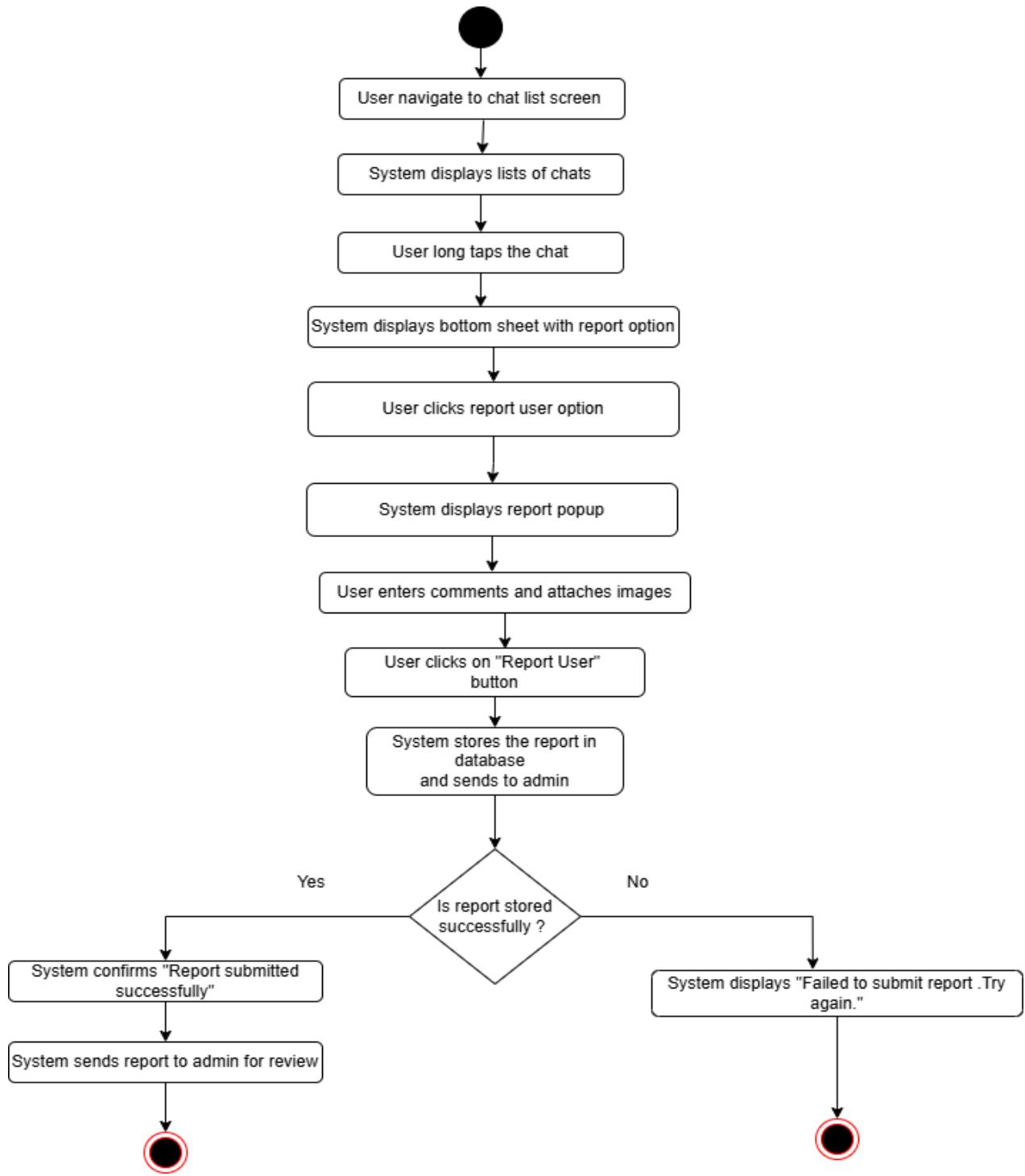
Use case 23: Provide feedback for tutor



Use case 24: Add Packages

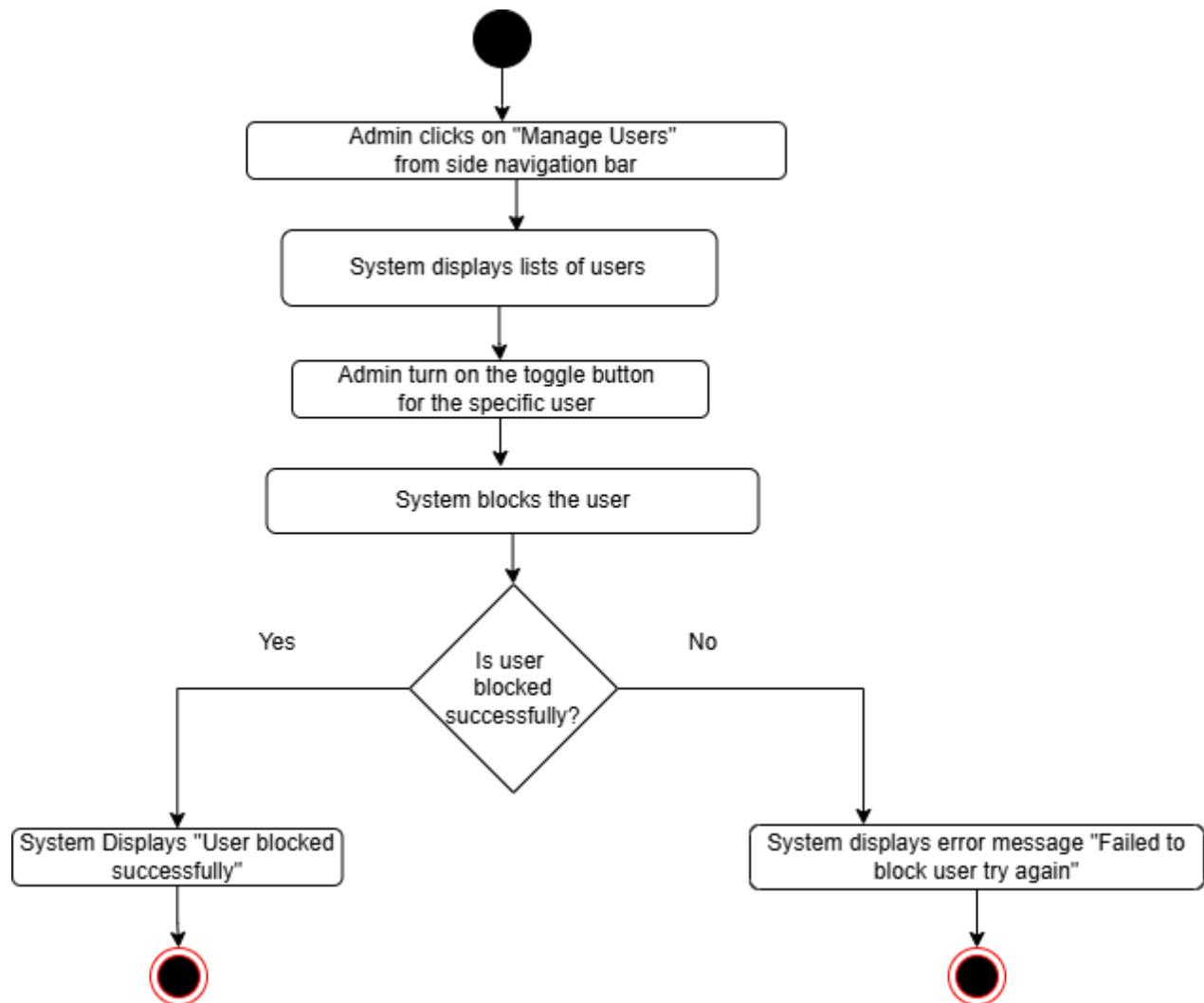


Use case 25: Report user

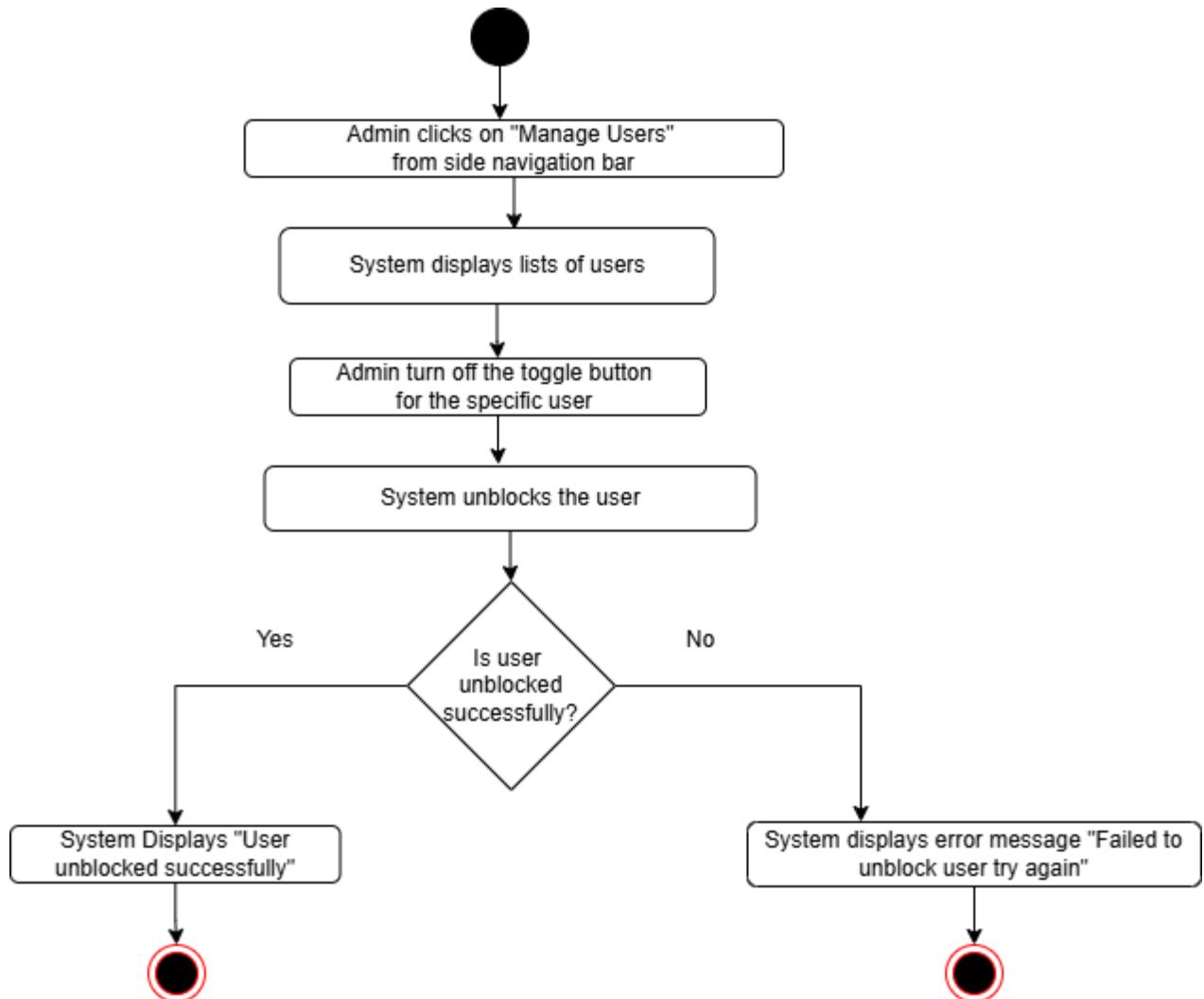


S

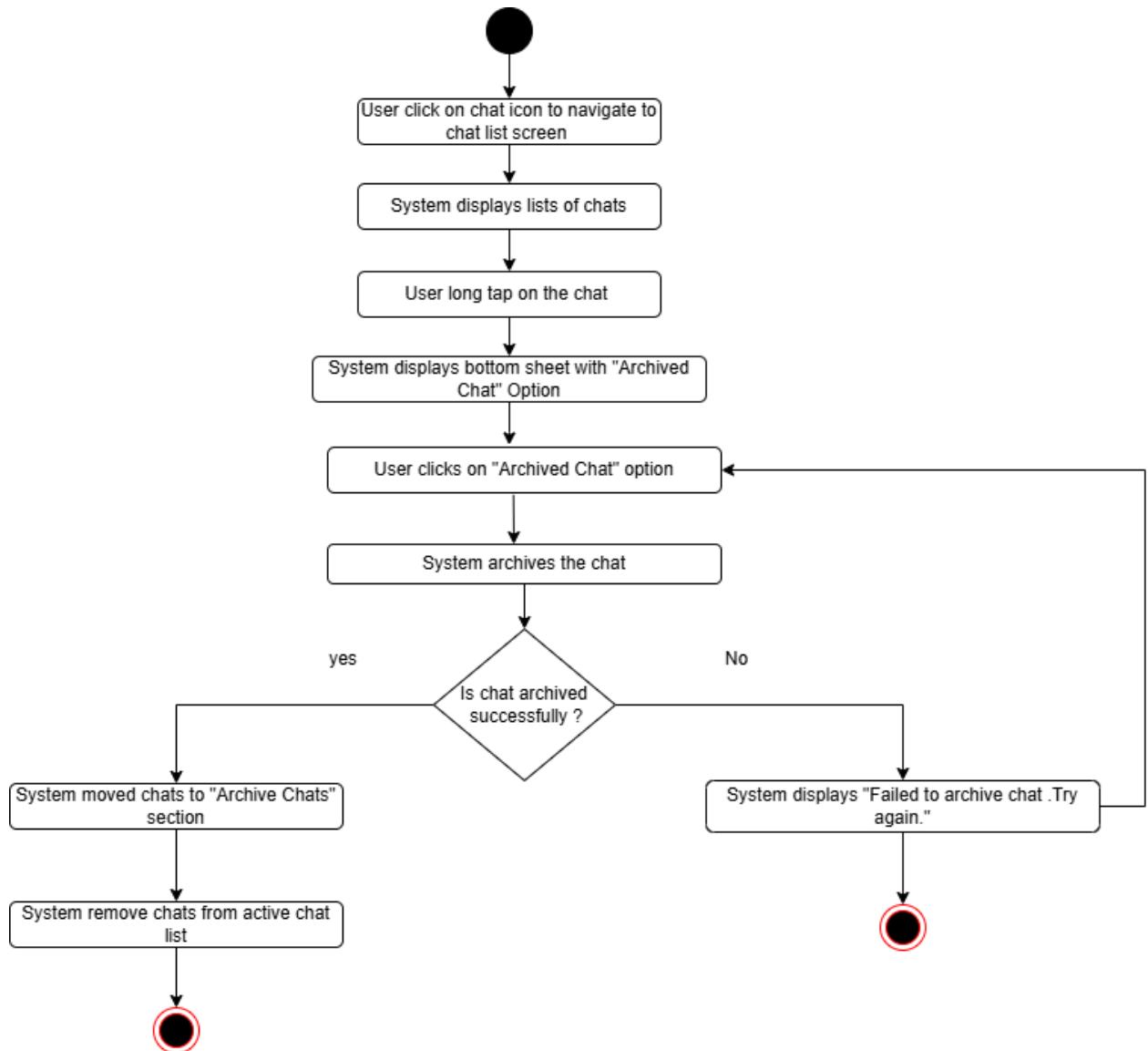
Use case 26: Block User



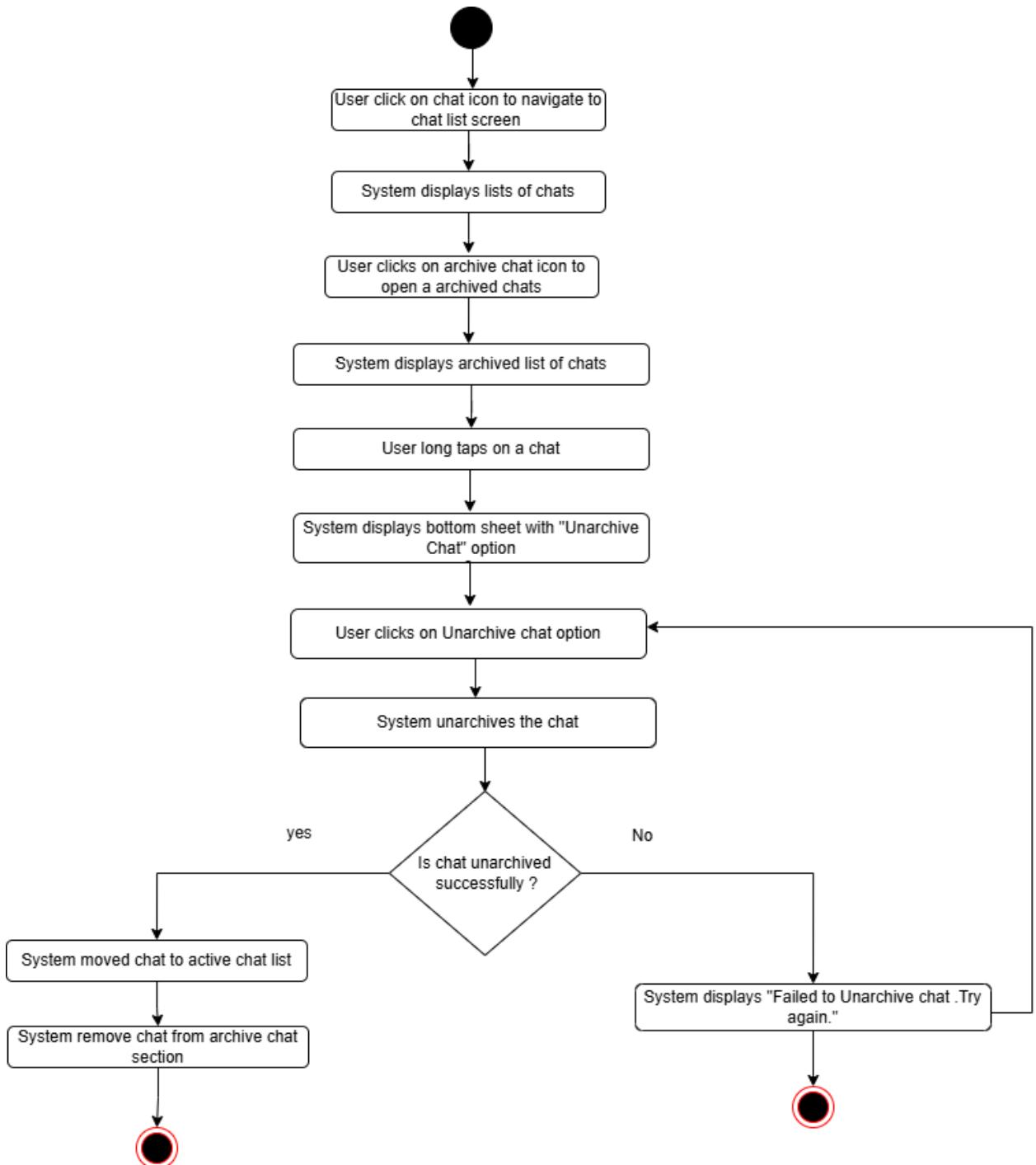
Use case 27: Unblock user



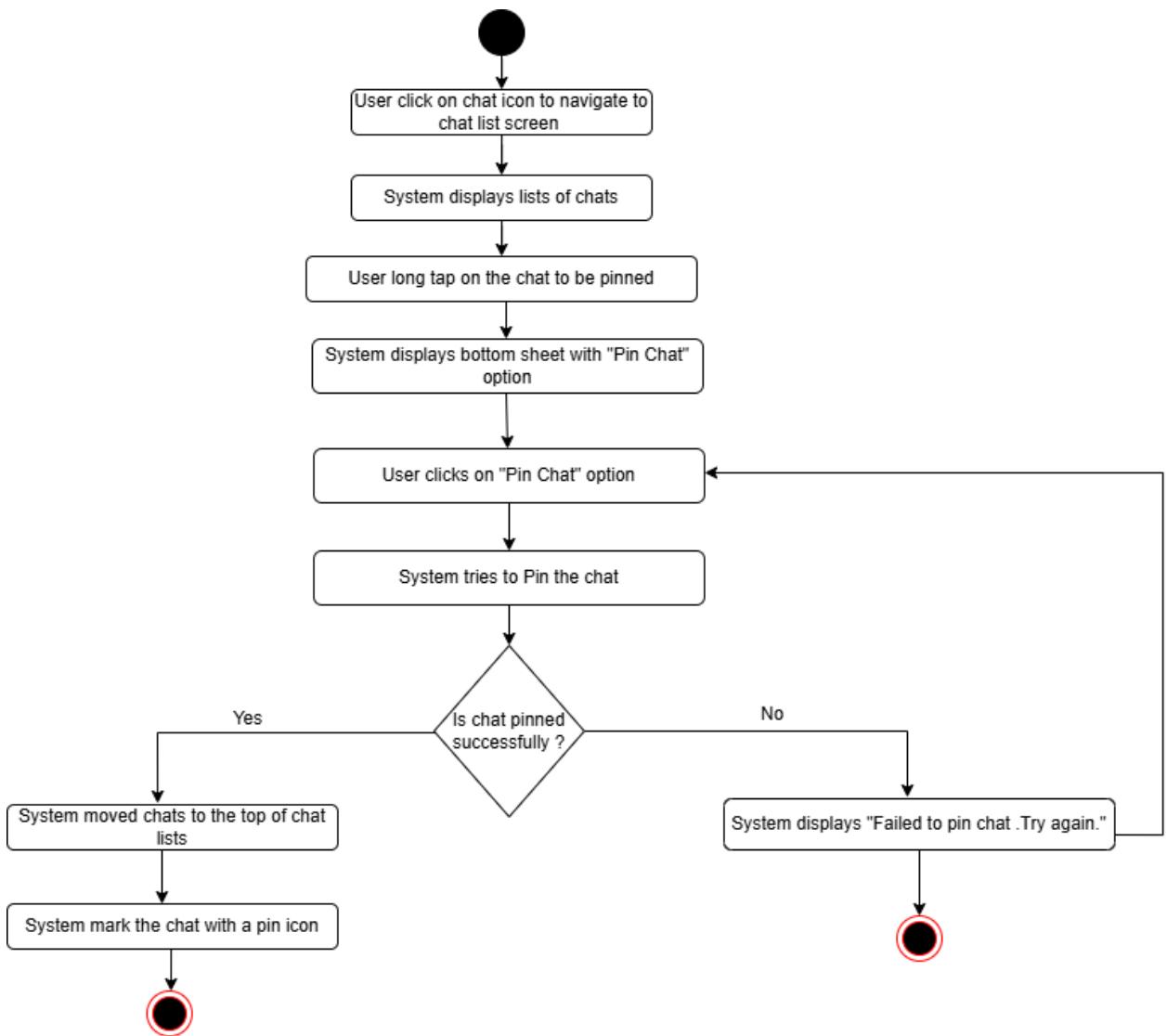
Use case 28: Archive chat



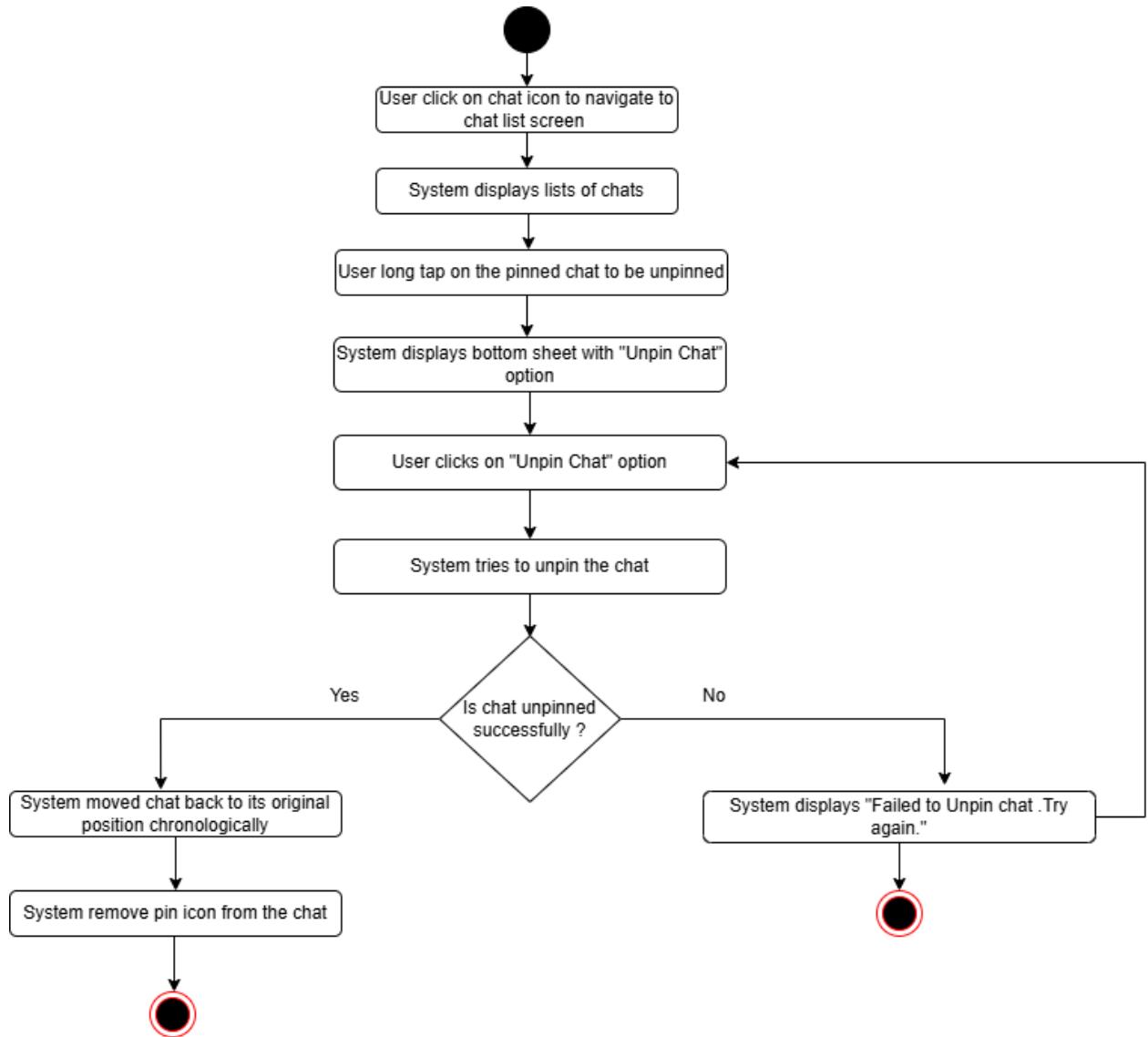
Use case 29: Unarchive chat



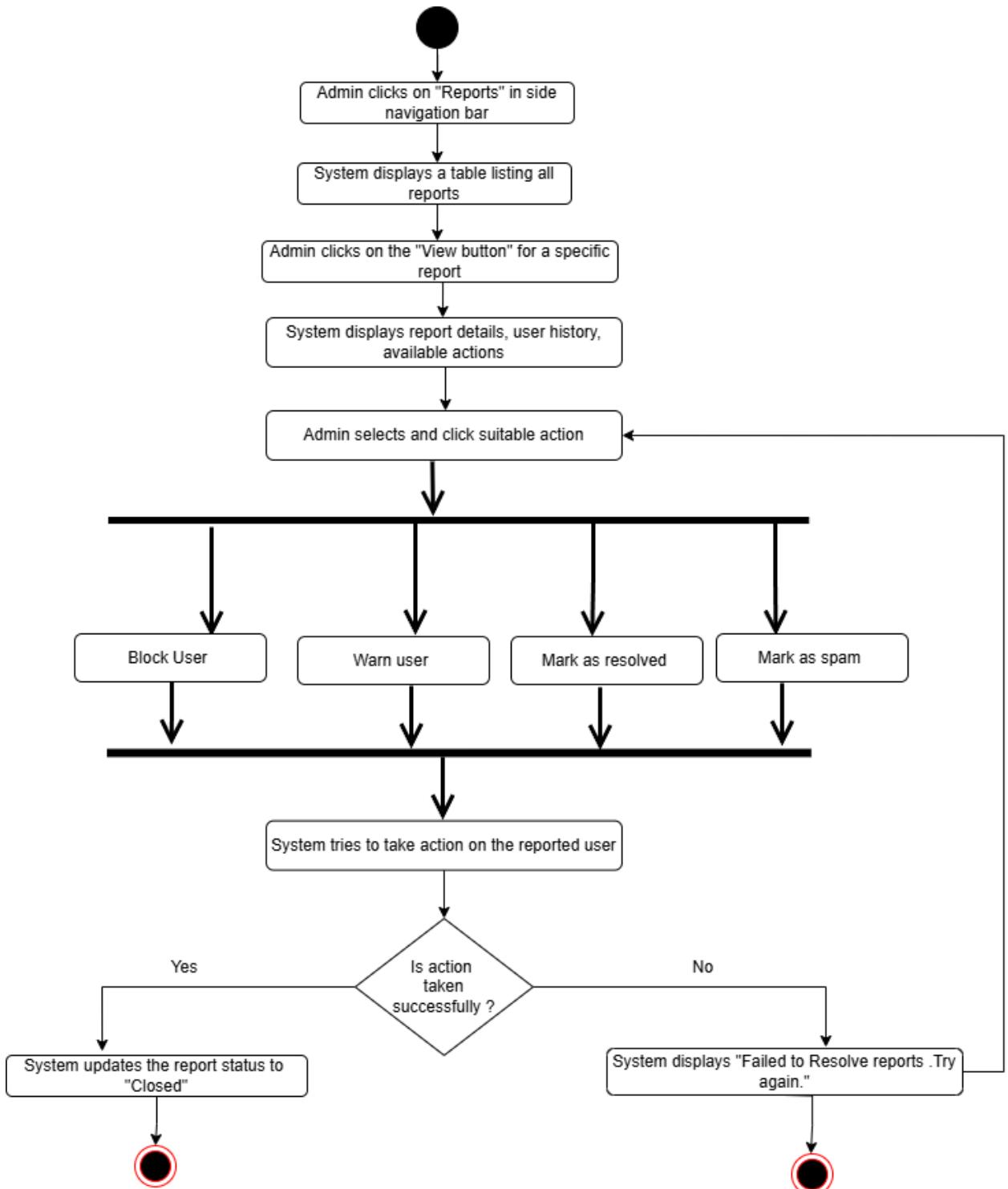
Use case 30: Pin chat



Use case 31: Unpin chat



Use case 32: Resolve reports



4.5 Black Box Testing

4.5.1 Register Account

4.5.1.1 Scenarios through combination of basic flow and alternate flow

Scenario	Flow
Scenario 1	Basic Flow
Scenario 2	Basic flow + A8.1 (Invalid email)
Scenario 3	Basic Flow + A8.2 (Invalid name)
Scenario 4	Basic Flow + A8.3 (Password Does Not Meet Requirements)
Scenario 5	Basic Flow + A8.4 (Password and Confirm Password Do Not Match)
Scenario 06	Basic Flow + A8.5(Missing field)

4.5.1.2 Inputs, Valid & Invalid Classes (ECP)

Input	Valid	Invalid
User type	{Student, Parent, Tutor}	{Empty (No selection)}
Name	{2–50 characters, A-Z, a-z}	{< 2 characters, > 50 characters}
Email	example@domain.com	{Email! = example@domain.com}, {user@, @domain.com, userdomain.com, user@domain, user@@domain.com, .user@domain.com, user@example.com}
Password	{Length \geq 8, 1 uppercase, 1 number, 1 special character}	{Length < 8, no uppercase, no number, no special character}
Confirm Password	Confirm Password = Password	Confirm Password != Password

4.5.1.3 Test Case Design

Test Case ID	Scenario	User Type	Email	Name	Password	Confirm Password	Expected Outcome	Actual Output
TC1	Scenario 1	Student	ahmed@gmail.com	Ahmed	Ahmed Abbas 123.	AhmedAbbas 123.	Account registered successfully	“Registration successful”
TC2	Scenario 2	Parent	ahmed@@gmail.co m	Ahmed	Ahmed abbas1 23.	AhmedAbbas 123.	invalid email	“The field is not a valid email address”
TC3	Scenario 3	Tutor	ahmer@gmail.com	A	Ahmer bilal12 3.	Ahmerbilal12 3.	invalid name	“Full Name must be at least 2 characters long”
TC4	Scenario 4	Student	ahmer@gmail.com	Ahmer	Ahmer 123	Ahmer123	Password does not meet requirement .	“Password must contain at least one special character”
TC5	Scenario 5	Parent	aliyan@gmail.com	Aliyan	Aliyan rizvi12 3.	Aliyanrizvi1.	Passwords do not match. Please try again.	“Confirm password does not match”
TC6	Scenario 6	Student	aliyan@gmail.com	Aliyan	Aliyan rizvi12 3.		Confirm password field is missing.	“Confirm password does not match”

4.5.2 Login Account

4.5.2.1 Scenarios through combination of basic flow and alternate flow

Scenario	Flow
Scenario 1	Basic Flow
Scenario 2	Basic flow + A5 (Invalid email or password)
Scenario 3	Basic flow + A5 (Invalid email or password)

4.5.2.2 Inputs, Valid & Invalid Classes (ECP)

Input	Valid	Invalid
Email	{registered email}	{unregistered email}
Password	{correct password for entered email}	{Incorrect password for entered email}

4.5.2.3 Test Case Design

Test Case ID	Scenario	Email	Password	Expected Output	Actual Output
TC1	Scenario 1	ahmedabbas8811@gmail.com (registered email)	Ahmedabbas123. (correct password)	User successfully logged in	Logged in without showing a successful prompt
TC2	Scenario 2+3	ahmedabbas88@gmail.com (unregistered email)	Ahmedabbas. (incorrect password)	Invalid email or password	"An error occurred during login"
TC3	Scenario 2	ahmedabbas88@gmail.com (unregistered email)	Ahmedabbas123. (correct password)	Invalid email or password	"An error occurred"

					during login"
TC4	Scenario 3	ahmedabbas8811@gmail.com (registered email)	Ahmed123(incorrect password)	Invalid email or password	"An error occurred during login"

4.5.3 Add Bio

4.5.3.1 Scenarios through combination of basic flow and alternate flow.

Scenario	Flow
Scenario 1	Basic Flow
Scenario 2	Basic flow + A4.1 (Bio field missing)
Scenario 3	Basic flow + A4.2 (Exceeding character limit)

4.5.3.2 Inputs, Valid & Invalid Classes (ECP).

Input	Valid	Invalid
Bio	{1–150 characters}	{Empty, >150 characters}

4.5.3.3 Test Case Design

Test Case ID	Scenario	Bio	Expected Output	Actual Output
TC1	Scenario 1	"I am a passionate tutor with 5 years of experience in math and science, helping students achieve academic success through personalized guidance."	Bio saved successfully	Navigated to next incomplete step
TC2	Scenario 2	Missing	"Bio field is empty."	"Please enter bio"

TC3	Scenario 2	"I am an experienced tutor specializing in math, science, and English. With over 10 years of teaching, I aim to help students excel academically through tailored lesson plans and a hands-on approach to problem-solving, ensuring their success."	"Bio must not exceed 150 characters."	Not allowing to enter more than 150 characters
------------	------------	---	---------------------------------------	--

4.5.4 Add Location

4.5.4.1 Scenarios through combination of basic flow and alternate flow.

Scenario	Flow
Scenario 1	Basic Flow
Scenario 2	Basic flow + A4.1 (Province not selected)
Scenario 3	Basic flow + A4.2 (City not selected)

4.5.4.2 Inputs, Valid & Invalid Classes (ECP).

Input	Valid	Invalid
Location	{Punjab, Sindh, etc.}	{Empty (Not Selected)}
City	{Lahore, Karachi, etc.}	{Empty (Not Selected)}

4.5.4.3 Test Case Design

Test Case ID	Scenario	Province	City	Expected Output	Actual Output
TC1	Scenario 1	Punjab	Lahore	Location saved successfully	Navigated to next incomplete step
TC2	Scenario 2	Empty	-	Province not selected	“Please select a province”
TC3	Scenario 3	Punjab	Empty	Please select a city.	“Please select a city”

4.5.5 Add Image

4.5.5.1 Scenarios through combination of basic flow and alternate flow.

Scenario	Flow
Scenario 1	Basic Flow
Scenario 2	Basic flow + A6.1 (Image not selected)
Scenario 3	Basic flow + A6.2 (Invalid image type)
Scenario 4	Basic flow + A6.3 (Invalid image size)

4.5.5.2 Inputs, Valid & Invalid Classes (ECP).

Input	Valid	Invalid
Image type	{JPG, PNG}	{!= JPG, PNG}
Image size	{≤ 1 MB}	{> 1 MB}

4.5.5.3 Test Case Design

Test Case ID	Scenario	Image type	Image size	Expected Output	Actual Output
TC1	Scenario 1	JPG	1 MB	Image successfully uploaded	Navigated to next incomplete step
TC2	Scenario 1	PNG	1 MB	Image successfully uploaded	Navigated to next incomplete step.
TC3	Scenario 2	(Not selected)	(Not selected)	Please upload an image	“Please select image”
TC4	Scenario 3	GIF	1 MB	Invalid image type	Navigated to next incomplete step.
TC5	Scenario 4	PNG	5 MB	Invalid image size	“The selected file is too large please choose a file smaller than 1 MB.”

4.5.6 Add CNIC

4.5.6.1 Scenarios through combination of basic flow and alternate flow.

Scenario	Flow
Scenario 1	Basic Flow
Scenario 2	Basic flow + A6.1 (file not selected)
Scenario 3	Basic flow + A6.2 (Invalid file type)
Scenario 4	Basic flow + A6.3 (Invalid file size)

4.5.6.2 Inputs, Valid & Invalid Classes (ECP).

Input	Valid	Invalid
Image type	{, pdf}	{!= pdf}
Image size	{≤2 MB}	{> 2MB}

4.5.6.3 Test Case Design

Test Case ID	Scenario	Image type	Image size	Expected Output	Actual Output
TC1	Scenario 1	pdf	1 MB	file successfully uploaded	Navigated to next incomplete step
TC2	Scenario 2	(Not selected)	(Not selected)	Please upload cnic file	“Please upload cnic”
TC3	Scenario 3	docx	3 MB	Invalid file type	“Please upload cnic”
TC4	Scenario 4	pdf	4 MB	Invalid file size	“Max file size is 2 MB”

4.5.7 Add Qualification

4.5.7.1 Scenarios Through Combination of Basic Flow and Alternate Flow:

Scenario	Flow
Scenario 1	Basic Flow
Scenario 2	Basic Flow + A8.1 (Educational Level not selected)
Scenario 3	Basic Flow + A8.2 (Institute Name not entered)
Scenario 4	Basic Flow + A8.3 (No file selected)

Scenario 5	Basic Flow + A8.4 (Invalid file type selected)
Scenario 6	Basic Flow + A8.5 (Invalid file size selected)

4.5.7.2 Inputs, Valid & Invalid Classes (ECP):

Input	Valid Class	Invalid Class
Education Level	{Under matric, Matric, Intermediate, Bachelor's, Master's, PhD}	{Empty}
Institute Name	{1–20 characters, A-Z, a-z, 0-9}	{Empty, Special Characters, >20 chars}
File Type	{PDF}	{!=pdf}
File Size	{≤2 MB}	{> 2 MB}

4.5.7.3 Test Case Design:

Test Case ID	Scenario	Education Level	Institute Name	File Type	File Size	Expected Outcome	Actual output
TC1	Scenario 1	Bachelor's	XYZ University	Pdf	2 MB	Qualification saved successfully	Navigated to next incomplete step.
TC2	Scenario 2	-	XYZ University	Pdf	2 MB	Please select an education level.	"Please Fill All Fields "
TC3	Scenario 3	Master's	-	Pdf	1 MB	Please enter an institute name.	"Please Fill All Fields"

TC4	Scenario 4	PhD	ABC University			Please select a file to upload.	“Please attach your qualification document”
TC5	Scenario 5	Bachelor's	DEF Institute	docx	1 MB	Invalid file type. Please select a valid file type	“Only pdf files are allowed”
TC6	Scenario 6	Master's	XYZ University	Pdf	6 MB	Invalid file size.	“Max file size is 2 MB”

4.5.8 Add Teaching Experience

4.5.8.1 Scenarios Through Combination of Basic Flow and Alternate Flow:

Scenario	Flow
Scenario 1	Basic Flow
Scenario 2	Basic Flow + A9.1 (Missing educational level)
Scenario 3	Basic Flow + A9.2 (Missing start date)
Scenario 4	Basic Flow + A9.3 (Missing end date when "I still work here" is toggled off)
Scenario 5	Basic Flow + A9.4 (No file selected)
Scenario 6	Basic Flow + A9.5 (Invalid file type selected)
Scenario 7	Basic Flow + A9.6 (Invalid file size selected)

4.5.8.2 Inputs, Valid & Invalid Classes (ECP):

Input	Valid Class	Invalid Class
Education Level of students	{Under matric, Matric, Intermediate, Bachelor's, Master's, PhD}	{Empty}
Start date	{start date before end date}	{Empty, start date after end date}

End date	{end date after start date, Selected (when toggle button off)}	{Empty, end date before start date, NotSelected (when toggle button on)}
File Type	{PDF}	{!=pdf}
File Size	{≤2 MB}	{> 2 MB}

4.5.8.3 Test Case Design:

Test Case ID	Scenario	Education Level of Students	Start date	End date	Still Work here	File Type	File Size	Expected Outcome	Actual Output
TC1	Scenario 1	Matric	12/01/2022	12/08/2022	false	Pdf	1 MB	Experience saved successfully	Navigated to next screen without displaying 'Experience saved successfully' message.
TC2	Scenario 1	FSc	12/01/2021	-	true	Pdf	2 MB	Experience saved successfully	Navigated to next screen without displaying 'Experience saved successfully' message.
TC3	Scenario 2	-	12/01/2021	-	true	Pdf	2 MB	Please enter education level of students.	"Please select student education level!"
TC4	Scenario 3	Masters	-	12/08/2022		Pdf	2 MB	Missing start date	"Please enter start date"
TC6	Scenario 4	Bachelor's	12/01/2022	-	false	pdf	2 MB	Missing end date	"Please enter end date or

									mark as still working”
TC8	Scenario 5	Matric	03/06/2021	-	true	-	-	Select a file to upload	“Please upload proof of experience”
TC9	Scenario 6	FSc	03/06/2021	06/03/2022	false	.docx	2 Mb	In valid file type	“Only pdf files are allowed”
TC10	Scenario 7	FSc	03/06/2021	-	true	Pdf	6 Mb	Invalid file size	“Max file size is 2mb”

4.5.9 Change password

4.5.9.1 Scenarios Through Combination of Basic Flow and Alternate Flow

Scenario	Flow
Scenario 1	Basic Flow
Scenario 2	Basic Flow + A9.2 (Weak Password)
Scenario 3	Basic Flow + A9.3 (Password Mismatch)

4.5.9.2 Inputs, Valid & Invalid Classes (ECP)

Input	Valid Class	Invalid Class
New Password	{Length \geq 8, 1 uppercase, 1 number, 1 special character}	{Length < 8, no uppercase, no number, no special character}
Confirm Password	Confirm Password= Password	Confirm Password! = Password

4.5.9.3 Test Case Design

Test Case ID	Scenario	Old Password	New Password	Confirm Password	Expected Outcome	Actual Output
TC1	Scenario 1	Ahmerbilal123.	Awonbilal123.	Awonbilal123.	Password successfully updated.	"Password Changed Your password has been successfully updated."
TC2	Scenario 2	Bilalkhan123. (correct old password)	Ali123.	Ali123.	Password is too weak.	"Password must be at least 8 characters"
TC3	Scenario 3	Bilalkhan123. (correct old password)	Adilraza123.	Adilraza12.	New password and confirm password do not match.	"Password Mismatch The new passwords do not match."

4.5.10 Edit availability

4.5.10.1 Scenarios Through Combination of Basic Flow and Alternate Flow

Scenario	Flow
Scenario 1	Basic Flow
Scenario 2	Basic Flow + A4 (Invalid Time Slot)

4.5.10.2 Inputs, Valid & Invalid Classes (ECP)

Input	Valid Class	Invalid Class
Start Time	{1 <= Hour <= 12}, {1 <= Minutes <= 59}, {Start time earlier than end time}.	{1 >Hour > 12}, {1 >= Minutes >= 59}, {Invalid time format or later than End Time}
End Time	{1 <= Hour <= 12}, {1 <= Minutes <= 59}, {End time later than start time}.	{1 >Hour > 12}, {1 >= Minutes >= 59}, {Invalid time format or earlier than Start Time}

4.5.10.3 Test Case Design

Test Case ID	Scenario	Start Time	End Time	Expected Outcome	Actual Output
TC1	Scenario 1	09:00 AM	11:00 PM	Availability successfully updated.	“Availability updated successfully”
TC2	Scenario 2	02:00 PM	01:00 PM	Invalid start or end time	“Start time must be earlier than end time.”
TC3	Scenario 2	12:00 PM	13:61	Invalid time format.	“Enter a valid time.”
TC4	Scenario 2	03:00 PM	02:00 PM	Invalid start or end time	“End time must be later than start time.”

4.5.11 Add Packages

4.5.11.1 Scenarios Through Combination of Basic Flow and Alternate Flow

Scenario	Flow
Scenario 1	Basic Flow
Scenario 2	Basic Flow + A9.1 (Empty required fields)
Scenario 3	Basic Flow + A9.2 (Database failure)
Scenario 4	Basic Flow + A9.3 (Description > 150 chars)
Scenario 5	Basic Flow + A9.4 (Negative duration)
Scenario 6	Basic Flow + A9.5 (Minutes not in 0-59)
Scenario 7	Basic Flow + A9.6 (Sessions/week ≤ 0)
Scenario 8	Basic Flow + A9.7 (Number of weeks ≤ 0)
Scenario 9	Basic Flow + A9.8 (Price ≤ 0)

4.5.11.2 Inputs, Valid & Invalid Classes (ECP)

Input	Valid Class	Invalid Class
Package Name	Non-empty string	Empty string
Package Description	≤ 150 characters	> 150 characters
Duration (Hours, Mins)	Hours ≥ 0, 0 ≤ Mins ≤ 59	Hours < 0, Mins < 0 or Mins > 59
Sessions per Week	> 0	≤ 0
Number of Weeks	> 0	≤ 0
Price	> 0	≤ 0

4.5.11.3 Test Case Design

Test Case ID	Scenario	Package Name	Package Description	Hour	Minute	Session /Week	No. of Weeks	Price	Expected Outcome	Actual Output
TC1	Scenario 1	Standard	Introductory package for Algebra	1	30	3	4	2500	Package added successfully	“Package added successfully”
TC2	Scenario 2	(empty)	Fast-track Math revision	1	45	2	3	1500	Prompt to fill all fields	“Package name cannot be empty.”
TC3	Scenario 3	Basic	Foundation English course	2	15	2	5	2000	Show storage error	“Failed to add package.”
TC4	Scenario 4	Premium	Learning never ends; every day offers a chance to grow, improve, and discover something new. Stay curious, stay motivated, and keep moving forward.	1	0	2	2	1800	Show length error	“Package description cannot exceed 150 characters.”
TC5	Scenario 5	Summer Package	Intensive summer science crash course	-1	30	3	6	2200	Negative duration error	“Hours and minutes must be non-negative integers.”
TC6	Scenario 6	Skill Up	Learn coding fundamentals in 3 weeks	1	75	4	3	2700	Minutes out of range error	“Minutes should be between 0 and 59.”

TC7	Scenario 7	Weekly Fit	Weekly fitness coaching	1	20	0	5	300 0	Sessions/week must be > 0	“Sessions per week must be greater than zero.”
TC8	Scenario 8	Quick Learn	Quick revision of Computer Basics	2	0	2	0	120 0	Weeks must be > 0	“Number of weeks must be greater than zero.”
TC9	Scenario 9	Mega Plan	Full semester science preparation	1	30	3	6	0	Price must be > 0	“Price must be greater than zero.”

4.5.12 Report User

4.5.12.1 Scenarios Through Combination of Basic Flow and Alternate Flow

Scenario	Flow
Scenario 1	Basic Flow
Scenario 2	Basic Flow + A7.1 (System failure)
Scenario 3	Basic Flow + A7.2 (Comment field missing)
Scenario 4	Basic Flow + A7.4 (Images not attached)
Scenario 5	Basic Flow + A7.5 (Invalid image type)

4.5.12.2 Inputs, Valid & Invalid Classes (ECP)

Input	Valid Class	Invalid Class
Comments	{Non-empty comments}	{empty comments}
Images	Image type = {png, jpg}, {images attached}	Image type! = {png, jpg}, {images not attached}

4.5.12.3 Test Case Design

Test Case ID	Scenario	Comments	Images Attached	Expected Outcome	Actual Output
TC1	Scenario 1	User was using abusive language.	proof.png	Report added successfully	“Report submitted for Ahmed Abbas”
TC2	Scenario 2	Sending spam links frequently.	spam.jpg	Failed to submit report	“Failed to submit report”
TC3	Scenario 3	(Empty)	proof1.jpg	Comment is required to submit a report.	“Please provide details for your report”
TC4	Scenario 4	Sharing personal contact info.	(None)	Attach images to submit report	“Please attach at least one image”
TC5	Scenario 5	User shared harmful file.	script.exe	invalid file type	“Only JPG and PNG images are allowed”

4.5.13 Submit feedback

4.5.13.1 Scenarios Through Combination of Basic Flow and Alternate Flow

Scenario	Flow
Scenario 1	Basic Flow
Scenario 2	Basic Flow + A7.1 (System failure)
Scenario 3	Basic Flow + A7.2 (Rating not selected)
Scenario 4	Basic Flow + A7.3 (Comment field missing)

4.5.13.2 Inputs, Valid & Invalid Classes (ECP)

Input	Valid Class	Invalid Class
Rating	{selected}	{not selected}
Comments	{Non-empty comments}	{empty comments}

4.5.13.3 Test Case Design

Test Case ID	Scenario	Comments	Rating	Expected Outcome	Actual Output
TC1	Scenario 1	Amazing tutor	5	Feedback submitted	“Review submitted!”
TC2	Scenario 2	Great tutor	4	Failed to submit feedback	Error: Unable to submit feedback. Please try again.
TC3	Scenario 3	Very helpful tutor	Not selected	Select rating	“Please select a rating and write a review.”
TC4	Scenario 4	(Empty)	3	fill in the comment field	“Please select a rating and write a review.”

4.6 Usability testing

4.6.1 Usability Testing Report (Tutor Perspective)

Test Case ID	Task Description	Goal	Success Criteria	Metrics to Measure
TUT01	Register and complete profile	Tutor should be able to create a profile easily	Profile setup successfully	Time: 4 min Errors: 3 (file size error)
TUT02	Set availability slots	Tutor should manage time slots intuitively	Availability saved correctly	Time: 2.5 min Errors: 0
TUT03	Create a tutoring package	Tutor can define session details for booking	Package visible to students	Time: 1 min 30 sec
TUT04	Send messages	Ensure smooth communication with students	Messages are delivered	Time: 25 sec Errors: 0
TUT05	Accept or decline booking request	Accept or decline booking	Correct status update in app	Time: 40 sec Errors: 0
TUT06	Generate weekly progress report	Track and share student progress	Report saved and shared	Time: 2.5 min Errors: 0
TUT07	Edit Profile image	Update profile image	Changes reflected immediately	Time: 1 min Errors: 0

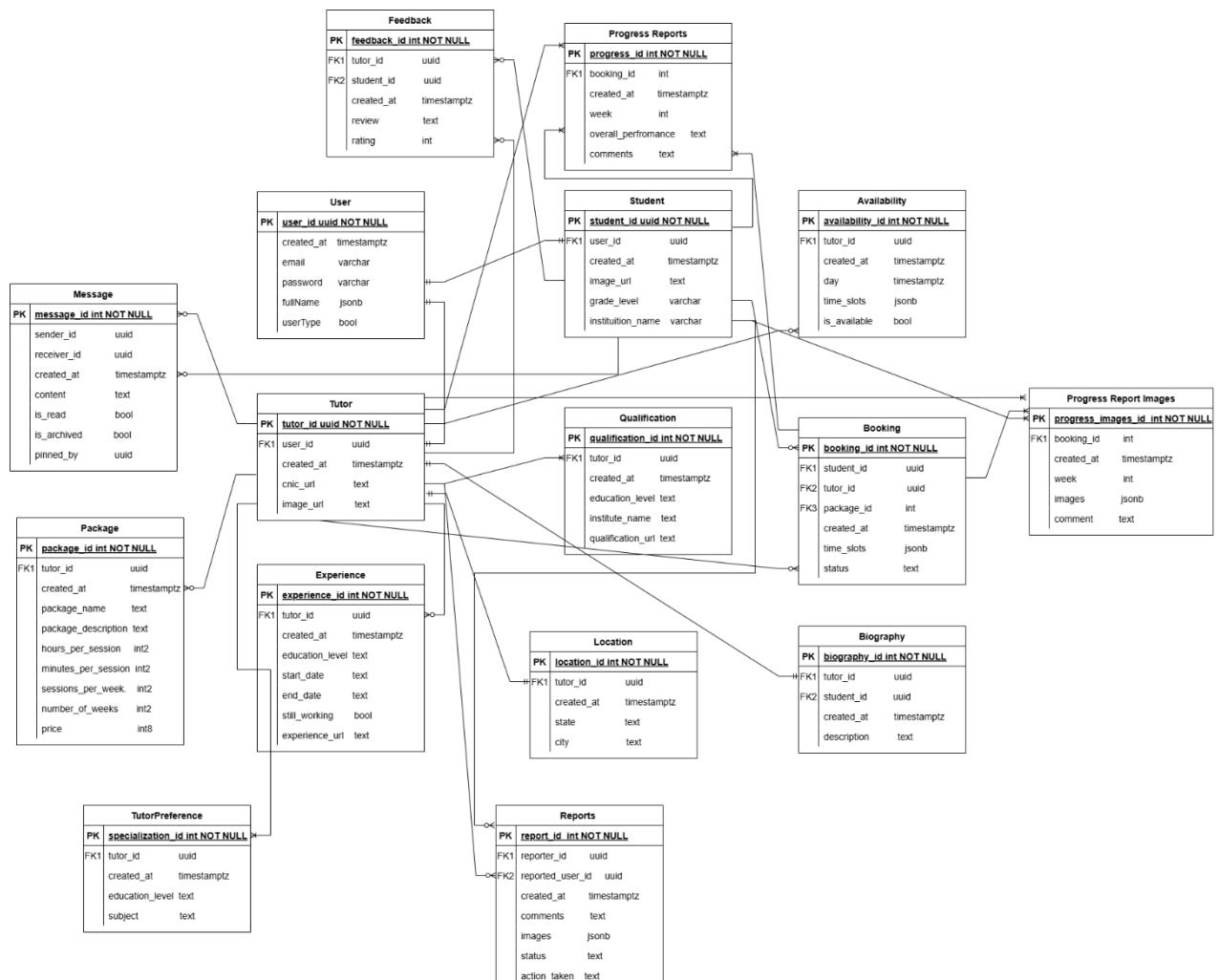
TUT09	Report a user	Raise issues with a student or parent	Report submitted to admin	Time: 1 min Error: 0
--------------	---------------	---------------------------------------	---------------------------	---------------------------------------

4.6.2 Usability Testing Report (Student Perspective)

Test Case ID	Task Description	Goal	Success Criteria	Metrics to Measure
UT01	Sign up as a new student	Account creation should be smooth	Account successfully created	Time: 1.5 min Errors: 1 (user entered weak password)
UT02	Search for a tutor by subject and city	Tutor discovery should be efficient	Relevant results appear correctly	Time: 50 sec Accuracy: High
UT03	Book a tutoring session	Booking should be fast and clear	Booking confirmation displayed	Time: 2 min Errors: 0
UT04	View and edit user profile	Update profile info without confusion	Updated info saved and reflected	Time: 1 min Errors: 0
UT06	Send messages to tutors	Messaging should feel real-time and reliable	Message appears in chat interface	Time: 30 sec Errors: 0
UT08	Provide feedback or rate a tutor	Give meaningful post-session feedback	Feedback submitted	Time: 1 min 20 sec Errors: 0

UT09	View Progress Report	See progress report	Progress report displayed for each week	Time: 30 sec Errors: 0
-------------	-----------------------------	----------------------------	--	---

4.7 ERD Diagram



Chapter 5:

Implementation

Chapter 5:

Implementation

5.1 Work Breakdown Structure

PART – 1

Week	Activity ID	Activity Name	Assigned To
1	1	Identify functional requirements	Ahmed Abbas + S.M Aliyan Rizvi
1	2	Identify non-functional requirements	Ahmed Abbas + S.M Aliyan Rizvi
1	3	Develop use cases	Ahmed Abbas
1	4	Develop fully dressed use cases	Ahmed Abbas + S.M Aliyan Rizvi
2	5	Create activity diagrams	Ahmed Abbas + S.M Aliyan Rizvi
2	6	Create sequence diagrams	Ahmed Abbas + S.M Aliyan Rizvi + M. Ahmer Bilal Jan
3	7	Create UI for sign-in and sign-up	M. Ahmer Bilal Jan
3	8	Implementation of UI for sign-in and sign-up	S.M Aliyan Rizvi
4	9	Develop backend for storing authentication data	Ahmed Abbas
4	10	Integrate database for registered user details	Ahmed Abbas

5	11	Document user authentication steps	Ahmed Abbas + S.M Aliyan Rizvi
5	12	Create UI for profile completion steps	M. Ahmer Bilal Jan
6	13	Implementation of UI for profile completion steps	S.M Aliyan Rizvi
6	14	Develop backend for storing profile data	M. Ahmer Bilal Jan
7	15	Integrate database for profile details	M. Ahmer Bilal Jan
7	16	Document profile completion steps	Ahmed Abbas
8	17	Write test cases for profile completion	Ahmed Abbas + S.M Aliyan Rizvi
8	18	Design UI for availability time slots	M. Ahmer Bilal Jan
9	19	Implementation of UI for availability time slots	S.M Aliyan Rizvi
9	20	Develop functionality to add/edit slots	Ahmed Abbas
11	21	Integrate with database	Ahmed Abbas
11	22	Write test cases for availability module	Ahmed Abbas + M. Ahmer Bilal Jan
12	23	Design UI for admin dashboard	M. Ahmer Bilal Jan

12	24	Implementation of UI for admin dashboard	S.M Aliyan Rizvi
12	25	Develop backend for verification logic	M. Ahmer Bilal Jan + Ahmed Abbas
12	26	Integrate admin approval system	M. Ahmer Bilal Jan
12	27	Write test cases for admin verification	Ahmed Abbas + S.M Aliyan Rizvi
12	28	Document test cases	Ahmed Abbas
12	29	Perform module-wise testing	Ahmed Abbas + M. Ahmer Bilal Jan

PART – 2

Week	Activity ID	Activity Name	Assigned To
1	1	Documentation of tutor profile	Ahmed Abbas
1	2	Testing of tutor profile	Ahmed Abbas
1	3	Design of tutor profile	M.Ahmer Bilal Jan
1	4	Frontend of tutor profile	S.M Aliyan Rizvi
1	5	Backend of tutor profile	M.Ahmer Bilal Jan + Ahmed Abbas
2	6	Documentation of search and filter	Ahmed Abbas + M.Ahmer Bilal Jan
2	7	Design of search and filter	M.Ahmer Bilal Jan
2	8	Frontend of search and filter	S.M Aliyan Rizvi

2	9	Backend of search and filter	M.Ahmer Bilal Jan
3	10	Documentation of tutor booking	Ahmed Abbas
3	11	Testing of tutor booking	Ahmed Abbas
3	12	Design of tutor booking	M.Ahmer Bilal Jan
3	13	Front end of tutor booking	S.M Aliyan Rizvi
3	14	Backend of tutor booking	Ahmed Abbas + M.Ahmer Bilal Jan
4	15	Backend of tutor availability in booking module	Ahmed Abbas + M.Ahmer Bilal Jan
5	16	Documentation of chat feature	Ahmed Abbas + S.M Aliyan Rizvi
5	17	Design of chat feature	M.Ahmer Bilal Jan
5	18	Frontend of chat feature	S.M Aliyan Rizvi
5	19	Backend of chat feature	Ahmed Abbas
6	20	Documentation of progress reports	Ahmed Abbas
6	21	Design of progress reports on tutor side	M.Ahmer Bilal Jan
6	22	Frontend of progress reports on tutor side	S.M Aliyan Rizvi
6	23	Backend of progress reports on tutor side	Ahmed Abbas + M.Ahmer Bilal Jan

7	24	Design of progress reports on student side	M.Ahmer Bilal Jan
7	25	Frontend of progress reports on tutor side	S.M Aliyan Rizvi
7	26	Backend of progress reports on tutor side	Ahmed Abbas
8	27	Design of admin dashboard	M.Ahmer Bilal Jan
8	28	Frontend of admin dashboard	S.M Aliyan Rizvi
9	29	Backend of admin dashboard	Ahmed Abbas
10	30	Documentation of manage users in admin	Ahmed Abbas + S.M Aliyan Rizvi
10	31	Design of manage users	M.Ahmer Bilal Jan
10	32	Front end of manage users	S.M Aliyan Rizvi
10	33	Backend of manage users	Ahmed Abbas
10	34	Documentation of submit reports	Ahmed Abbas + S.M Aliyan Rizvi
10	35	Design of submit reports	M.Ahmer Bilal Jan
10	36	Testing of submit reports	Ahmed Abbas
10	37	Frontend of submit reports	S.M Aliyan Rizvi
10	38	Backend of submit reports	Ahmed Abbas

11	39	Documentation of resolve reports in admin	Ahmed Abbas
11	40	Design of resolve reports	M.Ahmer Bilal Jan
11	41	Frontend of resolve report	S.M Aliyan Rizvi
11	42	Backend of resolve reports	Ahmed Abbas + M.Ahmer Bilal Jan
12	43	Documentation of submit feedback for tutor	Ahmed Abbas
12	44	Testing of submit feedback	Ahmed Abbas
12	45	Design of submit feedback	M.Ahmer Bilal Jan
12	46	Frontend of submit feedback	S.M Aliyan Rizvi
12	47	Backend of submit feedback	Ahmed Abbas
13	48	Documentation of student profile	Ahmed Abbas
13	49	Design of student profile	M.Ahmer Bilal Jan
13	50	Frontend of student profile	S.M Aliyan Rizvi
13	51	Backend of student profile	Ahmed Abbas
13	52	Deployment Diagram	M.Ahmer Bilal Jan

5.2 Components, Libraries, and APIs

Key technologies and libraries used in the development include:

Frontend Framework: Flutter for cross-platform mobile application development.

Backend: We used Supabase which is an open-source backend-as-a-service platform that provides a PostgreSQL database, authentication, real-time updates, file storage, and serverless functions to help developers build scalable and secure applications quickly.

State Management: We used GetX for state management, navigation, and implementing the MVC (Model-View-Controller) architecture to maintain a clean and organized code structure.

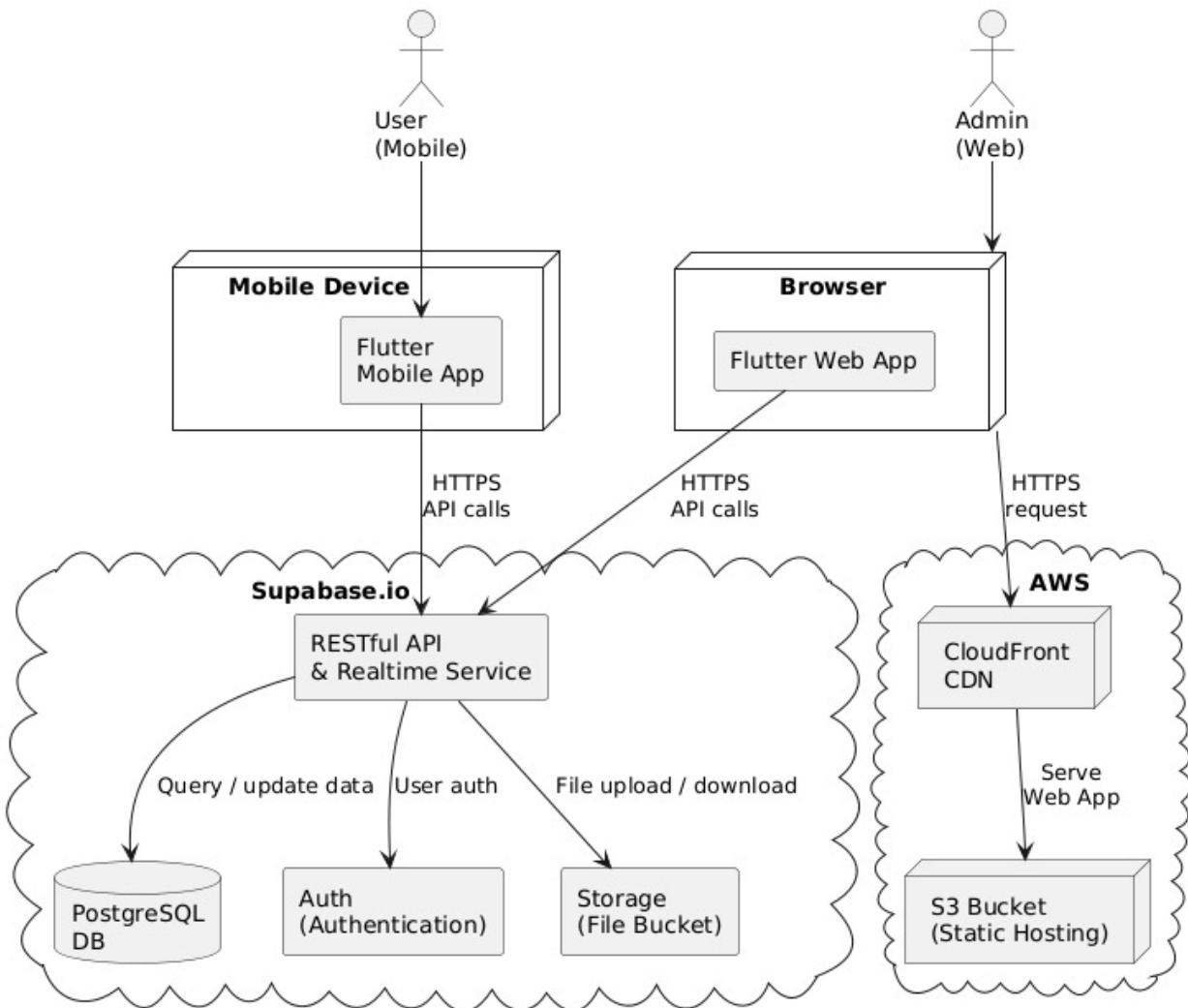
5.3 IDE, Tools, and Techniques

Figma: Used for designing and prototyping the app's UI.

VS Code: Used as the primary code editor for Flutter development.

GitHub: Used for version control, collaboration, and managing the project repository.

5.4 Deployment Diagram



Chapter 06:

Implementation

Chapter 6:

Conclusion and Outlook

6.1 Introduction

This chapter summarizes the achievements and outcomes of the TutorNow project, reflecting on the challenges faced, lessons learned, and the overall impact of the developed solution. It also provides a critical review of the project and outlines future recommendations to enhance the platform further.

6.2 Achievements and Improvements

The TutorNow project successfully addressed the identified gaps in the home tutoring market by providing a centralized platform for students, parents, and tutors. Key achievements include:

Streamlined Tutor Discovery:

Implemented search and filtering options to help students and parents find qualified tutors based on name, subject, educational level, qualification, price range.

Introduced a verified tutor system to ensure credibility and trust.

Efficient Communication:

Developed an in-app messaging feature to facilitate seamless communication between tutors, students, and parents, eliminating the need for external platforms.

Progress Tracking:

Enabled tutors to generate and share weekly progress reports, allowing parents to monitor their child's academic performance effectively.

Admin Oversight:

Created an admin dashboard for profile verification, report resolution, and platform management, ensuring a safe and reliable environment for all users.

These improvements have transformed the traditional tutoring process into a more efficient, transparent, and user-friendly experience.

6.3 Critical Review

While the project achieved its primary objectives, several challenges and limitations were encountered:

Technical Challenges:

- Integrating real-time communication, managing tutors availability and progress tracking required extensive testing to ensure reliability.
- Managing database performance for a growing user base posed scalability concerns.

User Adoption:

Encouraging tutors and students to transition from informal methods (e.g., social media groups) to a structured platform required effective onboarding and awareness campaigns.

Feature Limitations:

- The initial version lacked advanced analytics for tutors to track long-term student progress comprehensively.
- The review system could be enhanced with more detailed feedback mechanisms.
- Despite these challenges, the project demonstrated the viability of the solution and provided valuable insights for future enhancements.

6.4 Future Recommendations/Outlook

To further improve TutorNow, the following recommendations are proposed:

Advanced Analytics:

Introduce data-driven insights for tutors to analyze student performance trends and adjust teaching strategies accordingly.

Expanded Payment Options:

Support additional payment methods (e.g., mobile wallets, installment plans) to cater to a broader audience.

Enhanced Communication Tools:

Integrate video calling and screen-sharing features for virtual tutoring sessions.

Gamification:

Implement reward systems (e.g., badges, points) to motivate students and recognize top-performing tutors.

Scalability Improvements:

Optimize backend infrastructure to handle increased user traffic and ensure smooth performance.

Community Features:

Add forums or discussion boards where tutors and students can share resources and collaborate.

These enhancements will ensure TutorNow remains competitive and continues to meet the evolving needs of its users.

6.5 Summary

The TutorNow project successfully delivered a robust platform that bridges the gap between students, parents, and tutors. By addressing key pain points such as communication inefficiencies, payment delays, and lack of progress tracking, the platform has laid a strong foundation for future growth. While challenges were encountered, they provided valuable learning opportunities and highlighted areas for improvement. With continued development and user feedback, TutorNow has the potential to become a leading solution in the home tutoring market, fostering academic success and creating meaningful connections between educators and learners.