



COMSATS University Islamabad (CUI)

Assignment-05

CLO-4

Test Plan Document
for

Project Title
ROOMIE FIND

Version 1.0

Submitted By:

Zeeshan Qureshi FA24-BSE-078

Muhim Akhtar FA24-BSE-080

BSE-2-B

Supervisor

Mr. Tehseen Riaz Abbasi

Bachelor of Science in Software Engineering (2024-2028)

Table of Contents

1	Introduction	3
1.1	Document Purpose.....	3
1.2	Product Scope:	4
1.2.1	Brief Description	4
1.2.2	Product Features – RoomieFind	4
4	Executive Summary	5
4.1	Features to Be Tested (Revised with Modules)	6
4.2	Features Not to Be Tested (Aligned with Modules)	11
5	Testing and Evaluation	13
6	Test Cases	14
6.1	Unit Testing (Test cases)	14
6.2	Integration Testing:	67
	Module 1: User Authentication	67
	Module 2: Roommate Compatibility Assessment	68
	Module 3: Property and Room Listings	69
	Module 4: AI-based Matching Algorithm.....	69
	Module 5: Review & Rating System	70
	Module 6: Notification & Alert System.....	70
	Module 7: Admin Dashboard & Analytics	71
7	Module 8: Secure Chat & Agreement Templates	71
	System Testing:	72
8	Acceptance Test Cases – RoomieFind	74
9	Acceptance Testing Strategy – RoomieFind.....	76
9.1	Types of Acceptance Testing	76
9.2	Mapping with Generic Testing Types	76
10	Test Deliverables	77
6.	Conclusion.....	77
7.	Appendices.....	78

8.References	79
9.Paligirism Report.....	79

1 Introduction

The Software Test Case Document (STCD) outlines the testing strategy and detailed test cases for the *Roomie Find* platform, an amazing modern solution to your roommate related problems.

Purpose: The purpose of this website is to connect people that need roommates with similar interests and lifestyles.

- Key Goals: Providing the users with an easy-to-use matchmaking platform for those in need of roommates. Provide a simple but effective solution Promote responsible co-living
- Local Impact: 1.Decrease in the number of roommates that have stark differences among them
- Professional Impact: Enable partnerships between shelters and veterinarians to improve pet adoption outcomes.

1.1 Document Purpose

The purpose of this test case document is to define a structured approach for verifying and validating the functional behavior of the Roomie Find platform. It provides a detailed set of test cases to ensure that each module of the application operates as per the requirements detailed in SRS (Software Requirements Specification). This document serves as a guide for the quality assurance team .

1.2 Product Scope:

The Elder Care system is a web and mobile application developed to connect senior citizens with the verified caregivers and volunteers. Its purpose is to facilitate independent living and to improve the quality of life for the elderly.

1.2.1 Brief Description

The goal of this software is to provide a user-friendly and adequate solution for the people that are in search of roommates that match their vibe. This will not only make their living experience better but also will give the users a sense of harmony as they would be able to live their life according to their preferred living patterns with an understanding partner(roommate).

1.2.2 Product Features – RoomieFind

The RoomieFind software product will offer several integrated subsystems including user authentication, AI-based roommate and property matching, quiz-based personality assessment, property and room listings, secure chat with agreement templates, review and rating system, real-time notifications and alerts, and admin dashboard with analytics. These features are organized into functional modules to support students and young professionals in finding compatible roommates and suitable housing in a streamlined, secure, and user-friendly manner.

The system will not handle real estate transactions, provide legal landlord-tenant mediation, or integrate with third-party real estate platforms in real-time. It also does not include live property tour scheduling or advanced credit checks. Instead, RoomieFind focuses on fostering compatibility through data-driven matching, facilitating communication and agreement between users, and simplifying the process of shared accommodation discovery and management.

2 Application and Benefits – RoomieFind

Key Functionalities:

- AI-based roommate and property matching based on user preferences and quiz results.
- Personality assessment quiz to enhance compatibility scoring.
- Verified property and room listings with advanced search filters.
- Secure in-app chat system with pre-built agreement templates for smooth coordination.
- Review and rating system for roommates and properties to build community trust.
- Real-time notifications and alerts for new matches, messages, and system updates.
- Admin dashboard with analytics to monitor system performance and user activity.
- Profile customization and preference settings for tailored results.

Objectives:

- Help users find compatible roommates and housing options quickly and easily.
- Reduce roommate conflicts by promoting data-driven compatibility.
- Ensure secure communication and agreement management between matched users.
- Foster a trusted user community through reviews and verified listings.

Benefits:

- User-friendly design is optimized for students and young professionals on both web and mobile platforms.
- Increased safety and transparency through verified profiles and property listings.
- Quick and efficient roommate matching via AI and quiz-based systems.
- Streamlined communication and agreement processes for better coordination.

Limitations (What it will not do):

- Will not handle real estate transactions or payments directly.
- Does not offer legal mediation or tenant-landlord dispute resolution.
- Will not integrate with third-party real estate platforms in real time.
- Does not include live video tours or background/credit checks.

.

3 Intended Audience – RoomieFind

This document is intended for:

- **Client:** To verify that all the desired features and requirements of the RoomieFind platform are clearly defined, realistic, and aligned with the project goals.
- **Professor and Project Committee Members:** To evaluate the scope, clarity, feasibility, and completeness of the RoomieFind project as part of academic assessment criteria.
- **Developers:** To serve as a reference during the development phase, ensuring the system is implemented according to the specified functional and non-functional requirements.
- **Testers:** To design, execute, and validate test cases based on the documented system behaviors and ensure quality assurance throughout the development cycle.
- **Documentation Writers:** To create user manuals, FAQs, and training materials that accurately reflect RoomieFind's functionality and user experience.

4 Executive Summary

Testing will cover all core modules including user authentication, AI-based roommate and property matching, secure chat with agreement templates, review and rating system, and notifications. Due to academic resources and time constraints, testing will primarily focus on critical use cases and major system workflows that directly impact user experience and system reliability. Testing activities will be integrated with other evaluation processes such as requirement validation, code reviews, and user feedback sessions. Coordination among team

members will be maintained through regular meetings and communication tools to ensure seamless alignment between development and testing efforts.

4.1 Features to Be Tested (Revised with Modules)

The following modules and their key features will be tested from the user's perspective, categorized by risk level:

Feature	Test Design Specification (Module)	Description	Risk Level
User Registration	Module 1	Allows users to create an account using valid credentials.	High
User Login	Module 1	Authenticates user with username and password.	High
Password Reset	Module 1	Allows users to securely reset forgotten passwords.	High
Two-Factor Authentication	Module 1	Adds an extra layer of authentication via OTP/email.	High
Session Management	Module 1	Handles user session control and timeout logic.	High
Email Verification	Module 1	Verifies the authenticity of user's email address.	High
CAPTCHA Integration	Module 1	Ensures bot protection during registration/login.	High

Assignment 5 – Software Engineering (CSC-291)

User Role Management	Module 1	Assigns and manages different roles (e.g., admin, user).	High
Personality Questionnaire	Module 2	Collects personality data to aid in matching.	High
Lifestyle Preferences Input	Module 2	Collects lifestyle details like smoking, noise tolerance.	High
Daily Schedule Input	Module 2	Users input their daily schedule to compare compatibility.	High
Compatibility Score Calculation	Module 2	Generates a match score using user data.	High
Profile Matching Based on Scores	Module 2	Matches users with similar compatibility scores.	High
Match Feedback Collection	Module 2	Users provide feedback on suggested matches.	High
Incompatibility Alerts	Module 2	Alerts users about major mismatches.	High
Compatibility History	Module 2	Allows users to review past compatibility scores.	High
Post Property Listing	Module 3	User uploads room/property details.	High

Assignment 5 – Software Engineering (CSC-291)

Edit Property Listing	Module 3	User updates previously listed property info.	High
Search Listings	Module 3	Users search for properties based on criteria.	High
Filter Listings	Module 3	User filters results (location, price, availability).	High
View Property Details	Module 3	Displays full details of a selected listing.	High
Save Favorite Listings	Module 3	User saves listings to a favorites list.	High
Contact Property Owner	Module 3	User contacts owner via app or message.	High
Report Listing	Module 3	User flags an inappropriate or false listing.	High
Process Compatibility Inputs	Module 4	System reads inputs to determine matching logic.	High
Generate Match Scores	Module 4	Calculates match percentages based on inputs.	High
Suggest Top Matches	Module 4	Suggests users with highest compatibility.	High
Recalculate on Profile Update	Module 4	Updates match results when user info changes.	High

Assignment 5 – Software Engineering (CSC-291)

Filter Incompatible Profiles	Module 4	Removes profiles outside compatibility range.	High
AI Feedback Loop	Module 4	Uses interaction data to improve matching.	High
Update Match Algorithm	Module 4	Adjusts algorithm logic based on performance.	High
Log Match Outcomes	Module 4	Records matching success/failure events.	High
Report submission	Module 5	User submits reports visible to admin.	High
Rating in terms of stars	Module 5	User rates experiences with stars.	High
Editing a review	Module 5	Users can update their review content.	High
Reporting false/misleading reviews	Module 5	Users can report inappropriate reviews.	High
Send Notification	Module 6	System sends user/admin notification.	High
Set Notification Preferences	Module 6	User customizes how/when notifications are received.	High
Customize Notification Sound	Module 6	User sets custom sounds for notifications.	High

Assignment 5 – Software Engineering (CSC-291)

View Admin Activities	Module 7	Admin views records of system-wide activities.	High
Generate System Performance Report	Module 7	Admin generates system performance summaries.	High
Initiate Secure Chat	Module 8	User starts a secure conversation.	High
Send Encrypted Message	Module 8	Messages are encrypted end-to-end.	High
Upload Agreement Template	Module 8	Admin uploads reusable agreement formats.	High

4.2 Features Not to Be Tested (Aligned with Modules)

The following modules or features will not be tested in this release:

Module 1: User Authentication

Feature Name	Reason for Not Testing
User Role Management	Out of scope for basic testing, tested at admin level only
CAPTCHA Integration	Handled externally via third-party service; not tested internally

Module 2: Roommate Compatibility Assessment

Feature Name	Reason for Not Testing
Compatibility History	Feature not fully developed; pending backend logic
Incompatibility Alerts	Logic not finalized for current build

Module 3: Property and Room Listings

Feature Name	Reason for Not Testing
Report Listing	Admin-side review process incomplete
Save Favorite Listings	Functionality exists but integration is partial

Module 4: AI-based Matching Algorithm

Feature Name	Reason for Not Testing
Update Match Algorithm	Updates managed manually by dev team, not auto-triggered

AI Feedback Loop	Machine learning cycle disabled in test environment
------------------	-----------------------------------------------------

Module 5: Review & Rating System

Feature Name	Reason for Not Testing
Bulk review upload	Feature not yet developed
Real-time review sync across devices	Planned for future phase
Third-party integration for spam detection	Out of scope for current release

Module 6: Notification & Alert System

Feature Name	Reason for Not Testing
Customize Notification Sound	Dependent on user device settings
Mute Notifications	Planned for UX testing phase
Configure Notification Types	Feature not fully developed yet

Module 7: Admin Dashboard & Analytics

Feature Name	Reason for Not Testing
Download Analytics Report	Download testing handled separately
Monitor System Health	Planned for integration testing
Generate System Performance Report	Performance reports are part of load testing

Module 8: Secure Chat & Agreement Templates

Feature Name	Reason for Not Testing
Terminate Chat Session	Feature to be tested in later phases
Upload Agreement Template	Not yet implemented

View Agreement History	Covered in future maintenance release
------------------------	---------------------------------------

5 Testing and Evaluation

1. Verification

In the RoomieFind project, verification ensures that all features are built according to the functional and non-functional requirements. This involves checking if each module — such as Secure Chat, Agreement Templates, Review System, and Admin Analytics — has been correctly implemented based on the documented use cases and design specifications. For example, the backend logic in Python for sending encrypted messages or submitting a review must follow the exact flow and rules defined during the requirements phase.

2. Validation

Validation focuses on evaluating whether RoomieFind as a whole meets the actual needs of its users. This includes ensuring that the roommate matching process, secure communication, review submission, and agreement sharing features are useful, reliable, and aligned with the expectations of end users. For example, users should feel confident that the system matches them with suitable roommates and that their reviews or messages are processed securely.

3. Usability Testing

Usability testing is crucial to check how intuitive and user-friendly RoomieFind is. This includes assessing the layout and functionality of the user interfaces — such as the quiz for roommate preferences, the review submission form, or the secure chat window. Even though the backend is written in Python, its efficiency contributes to faster responses and smoother user experience, which are important for overall usability.

4. Unit Testing

Unit testing in RoomieFind targets individual functions and modules developed in Python. Functions like `match_users()`, `submit_review()`, `send_encrypted_message()`, or `notify_user()` are tested independently to ensure they work as expected in isolation. These tests help catch logical errors early in the development phase before they affect larger integrated systems.

5. Integration Testing

Once individual units are tested, integration testing ensures that various RoomieFind modules can interact with each other without issues. For instance, the system must be tested to verify that once users are matched by the AI algorithm, they can immediately

initiate a secure chat, share agreements, and leave reviews — all while the data flows smoothly between these interconnected modules.

6. System Testing

System testing is a comprehensive test of the full RoomieFind application. It checks if the system functions correctly from start to finish — from users taking a quiz and getting matched, to chatting, signing agreements, submitting reviews, and receiving notifications. This testing validates the overall behavior of the platform in a simulated production environment.

7. Regression Testing

As the project evolves and new features or updates are introduced, regression testing is performed to ensure that previously working functionalities remain unaffected. For example, if a new filter is added to the review system, it should not break existing features like editing or deleting reviews. Automated Python test scripts can help in consistently checking for regressions after every update.

8. Acceptance Testing

This is the final stage where RoomieFind is tested from the perspective of the client, instructor, or end users. Acceptance testing ensures that the application meets all project goals, functional requirements, and user expectations. Only after successful acceptance testing will RoomieFind be considered ready for final deployment or demonstration.

6 Test Cases

6.1 Unit Testing (Test cases)

Module 1: User Authentication

Test Case ID	TC 1.1
Test Case Name	User Registration
Preconditions	Users must be registered and logged in.
Test Steps	1. Login to the system. 2. Navigate to: 'User Registration'.

Assignment 5 – Software Engineering (CSC-291)

	3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• User Registration functions as expected
Actual Result	• User Registration functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-1
Module Reference	Module 1: User Authentication

Test Case ID	TC 1.2
Test Case Name	User Login
Preconditions	User must be registered and logged in.
Test Steps	1. Login to the system. 2. Navigate to: 'User Login'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• User Login functions as expected
Actual Result	• User Login functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025

Assignment 5 – Software Engineering (CSC-291)

Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-2
Module Reference	Module 1: User Authentication

Test Case ID	TC 1.3
Test Case Name	Password Reset
Preconditions	User must be registered and logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the system. 2. Navigate to: 'Password Reset'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Password Reset functions as expected
Actual Result	• Password Reset functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-3
Module Reference	Module 1: User Authentication

Test Case ID	TC 1.4
Test Case Name	Two-Factor Authentication

Preconditions	User must be registered and logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the system. 2. Navigate to: 'Two-Factor Authentication'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Two-Factor Authentication functions as expected
Actual Result	• Two-Factor Authentication functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-4
Module Reference	Module 1: User Authentication

Test Case ID	TC 1.5
Test Case Name	Session Management
Preconditions	User must be registered and logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the system. 2. Navigate to: 'Session Management'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data

Assignment 5 – Software Engineering (CSC-291)

Expected Result	• Session Management functions as expected
Actual Result	• Session Management functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-5
Module Reference	Module 1: User Authentication

Test Case ID	TC 1.6
Test Case Name	Email Verification
Preconditions	User must be registered and logged in.
Test Steps	1. Login to the system. 2. Navigate to: 'Email Verification'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Email Verification functions as expected
Actual Result	• Email Verification functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar

Assignment 5 – Software Engineering (CSC-291)

Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-6
Module Reference	Module 1: User Authentication

Test Case ID	TC 1.7
Test Case Name	CAPTCHA Integration
Preconditions	User must be registered and logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the system. 2. Navigate to: 'CAPTCHA Integration'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• CAPTCHA Integration functions as expected
Actual Result	• CAPTCHA Integration functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-7
Module Reference	Module 1: User Authentication

Test Case ID	TC 1.8
--------------	--------

Assignment 5 – Software Engineering (CSC-291)

Test Case Name	User Role Management
Preconditions	User must be registered and logged in.
Test Steps	1. Login to the system. 2. Navigate to: 'User Role Management'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• User Role Management functions as expected
Actual Result	• User Role Management functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-8
Module Reference	Module 1: User Authentication

Module 2: Roommate Compatibility Assessment

Test Case ID	TC 2.1
Test Case Name	Personality Questionnaire
Preconditions	User must be registered and logged in.
Test Steps	1. Login to the system. 2. Navigate to: 'Personality Questionnaire'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Personality Questionnaire functions as expected
Actual Result	• Personality Questionnaire functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-1
Module Reference	Module 2: Roommate Compatibility Assessment

Test Case ID	TC 2.2
Test Case Name	Lifestyle Preferences Input
Preconditions	User must be registered and logged in.
Test Steps	1. Login to the system. 2. Navigate to: 'Lifestyle Preferences

	Input'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Lifestyle Preferences Input functions as expected
Actual Result	• Lifestyle Preferences Input functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-2
Module Reference	Module 2: Roommate Compatibility Assessment

Test Case ID	TC 2.3
Test Case Name	Daily Schedule Input
Preconditions	User must be registered and logged in.
Test Steps	1. Login to the system. 2. Navigate to: 'Daily Schedule Input'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Daily Schedule Input functions as expected

Assignment 5 – Software Engineering (CSC-291)

Actual Result	• Daily Schedule Input functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-3
Module Reference	Module 2: Roommate Compatibility Assessment

Test Case ID	TC 2.4
Test Case Name	Compatibility Score Calculation
Preconditions	User must be registered and logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the system. 2. Navigate to: 'Compatibility Score Calculation'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Compatibility Score Calculation functions as expected
Actual Result	• Compatibility Score Calculation functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025

Assignment 5 – Software Engineering (CSC-291)

Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-4
Module Reference	Module 2: Roommate Compatibility Assessment

Test Case ID	TC 2.5
Test Case Name	Profile Matching Based on Scores
Preconditions	User must be registered and logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the system. 2. Navigate to: 'Profile Matching Based on Scores'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Profile Matching Based on Scores functions as expected
Actual Result	• Profile Matching Based on Scores functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-5

Assignment 5 – Software Engineering (CSC-291)

Module Reference	Module 2: Roommate Compatibility Assessment
------------------	---------------------------------------------

Test Case ID	TC 2.6
Test Case Name	Match Feedback Collection
Preconditions	User must be registered and logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the system. 2. Navigate to: 'Match Feedback Collection'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	<ul style="list-style-type: none"> • Match Feedback Collection functions as expected
Actual Result	<ul style="list-style-type: none"> • Match Feedback Collection functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-6
Module Reference	Module 2: Roommate Compatibility Assessment

Test Case ID	TC 2.7
Test Case Name	Incompatibility Alerts

Assignment 5 – Software Engineering (CSC-291)

Preconditions	User must be registered and logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the system. 2. Navigate to: 'Incompatibility Alerts'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Incompatibility Alerts functions as expected
Actual Result	• Incompatibility Alerts functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-7
Module Reference	Module 2: Roommate Compatibility Assessment

Test Case ID	TC 2.8
Test Case Name	Compatibility History
Preconditions	User must be registered and logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the system. 2. Navigate to: 'Compatibility History'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data

Assignment 5 – Software Engineering (CSC-291)

Expected Result	• Compatibility History functions as expected
Actual Result	• Compatibility History functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-8
Module Reference	Module 2: Roommate Compatibility Assessment

Module 3: Property and Room Listings

Test Case ID	TC 3.1
Test Case Name	Post Property Listing
Preconditions	User must be registered and logged in.
Test Steps	1. Login to the system. 2. Navigate to: 'Post Property Listing'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Post Property Listing functions as expected
Actual Result	• Post Property Listing functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-1
Module Reference	Module 3: Property and Room Listings

Test Case ID	TC 3.2
Test Case Name	Edit Property Listing
Preconditions	User must be registered and logged in.
Test Steps	1. Login to the system. 2. Navigate to: 'Edit Property Listing'.

	3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Edit Property Listing functions as expected
Actual Result	• Edit Property Listing functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-2
Module Reference	Module 3: Property and Room Listings

Test Case ID	TC 3.3
Test Case Name	Search Listings
Preconditions	User must be registered and logged in.
Test Steps	1. Login to the system. 2. Navigate to: 'Search Listings'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Search Listings functions as expected
Actual Result	• Search Listings functions as expected
Status	Pass
Created By	QA Team

Assignment 5 – Software Engineering (CSC-291)

Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-3
Module Reference	Module 3: Property and Room Listings

Test Case ID	TC 3.4
Test Case Name	Filter Listings
Preconditions	User must be registered and logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the system. 2. Navigate to: 'Filter Listings'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Filter Listings functions as expected
Actual Result	• Filter Listings functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-4
Module Reference	Module 3: Property and Room Listings

Test Case ID	TC 3.5
--------------	--------

Assignment 5 – Software Engineering (CSC-291)

Test Case Name	View Property Details
Preconditions	User must be registered and logged in.
Test Steps	1. Login to the system. 2. Navigate to: 'View Property Details'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• View Property Details functions as expected
Actual Result	• View Property Details functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-5
Module Reference	Module 3: Property and Room Listings

Test Case ID	TC 3.6
Test Case Name	Save Favorite Listings
Preconditions	User must be registered and logged in.
Test Steps	1. Login to the system. 2. Navigate to: 'Save Favorite Listings'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data

Assignment 5 – Software Engineering (CSC-291)

Expected Result	• Save Favorite Listings functions as expected
Actual Result	• Save Favorite Listings functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-6
Module Reference	Module 3: Property and Room Listings

Test Case ID	TC 3.7
Test Case Name	Contact Property Owner
Preconditions	User must be registered and logged in.
Test Steps	1. Login to the system. 2. Navigate to: 'Contact Property Owner'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Contact Property Owner functions as expected
Actual Result	• Contact Property Owner functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025

Assignment 5 – Software Engineering (CSC-291)

Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-7
Module Reference	Module 3: Property and Room Listings

Test Case ID	TC 3.8
Test Case Name	Report Listing
Preconditions	User must be registered and logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the system. 2. Navigate to: 'Report Listing'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Report Listing functions as expected
Actual Result	• Report Listing functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-8
Module Reference	Module 3: Property and Room Listings

Module 4: AI-based Matching Algorithm

Test Case ID	TC 4.1
Test Case Name	Process Compatibility Inputs
Preconditions	User must be registered and logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the system. 2. Navigate to: 'Process Compatibility Inputs'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Process Compatibility Inputs functions as expected
Actual Result	• Process Compatibility Inputs functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-1
Module Reference	Module 4: AI-based Matching Algorithm

Test Case ID	TC 4.2
Test Case Name	Generate Match Scores
Preconditions	User must be registered and logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the system. 2. Navigate to: 'Generate Match Scores'.

	3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Generate Match Scores functions as expected
Actual Result	• Generate Match Scores functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-2
Module Reference	Module 4: AI-based Matching Algorithm

Test Case ID	TC 4.3
Test Case Name	Suggest Top Matches
Preconditions	User must be registered and logged in.
Test Steps	1. Login to the system. 2. Navigate to: 'Suggest Top Matches'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Suggest Top Matches functions as expected
Actual Result	• Suggest Top Matches functions as expected

Assignment 5 – Software Engineering (CSC-291)

Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-3
Module Reference	Module 4: AI-based Matching Algorithm

Test Case ID	TC 4.4
Test Case Name	Recalculate on Profile Update
Preconditions	User must be registered and logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the system. 2. Navigate to: 'Recalculate on Profile Update'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Recalculate on Profile Update functions as expected
Actual Result	• Recalculate on Profile Update functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11

Assignment 5 – Software Engineering (CSC-291)

Use Case Reference	UC-4
Module Reference	Module 4: AI-based Matching Algorithm

Test Case ID	TC 4.5
Test Case Name	Filter Incompatible Profiles
Preconditions	User must be registered and logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the system. 2. Navigate to: 'Filter Incompatible Profiles'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Filter Incompatible Profiles functions as expected
Actual Result	• Filter Incompatible Profiles functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-5
Module Reference	Module 4: AI-based Matching Algorithm

Test Case ID	TC 4.6
Test Case Name	AI Feedback Loop
Preconditions	User must be registered and logged in.

Assignment 5 – Software Engineering (CSC-291)

Test Steps	1. Login to the system. 2. Navigate to: 'AI Feedback Loop'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• AI Feedback Loop functions as expected
Actual Result	• AI Feedback Loop functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-6
Module Reference	Module 4: AI-based Matching Algorithm

Test Case ID	TC 4.7
Test Case Name	Update Match Algorithm
Preconditions	User must be registered and logged in.
Test Steps	1. Login to the system. 2. Navigate to: 'Update Match Algorithm'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Update Match Algorithm functions as expected
Actual Result	• Update Match Algorithm functions as expected

Assignment 5 – Software Engineering (CSC-291)

Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-7
Module Reference	Module 4: AI-based Matching Algorithm

Test Case ID	TC 4.8
Test Case Name	Log Match Outcomes
Preconditions	User must be registered and logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the system. 2. Navigate to: 'Log Match Outcomes'. 3. Perform the operation. 4. Verify result.
Test Data	Valid sample input data
Expected Result	• Log Match Outcomes functions as expected
Actual Result	• Log Match Outcomes functions as expected
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Muhim Akhtar
Date of Execution	05/31/2025
Test Environment	Windows 11

Use Case Reference	UC-8
Module Reference	Module 4: AI-based Matching Algorithm

Module 5: Review & Rating System

This module is used for reviews and rating system so that feedback can be used to improve the application.

Test Case ID	TC 5.1
Test Case Name	Report submission
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Report submission'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Report submission works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Report submission works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-1
Module Reference	Module 5: Review & Rating System

Assignment 5 – Software Engineering (CSC-291)

Test Case ID	TC 5.2
Test Case Name	Rating in terms of stars
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Rating in terms of stars'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Rating in terms of stars works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Rating in terms of stars works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-2
Module Reference	Module 5: Review & Rating System

Test Case ID	TC 5.3
Test Case Name	Editing a review
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Editing a review'.

Assignment 5 – Software Engineering (CSC-291)

	3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Editing a review works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Editing a review works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-3
Module Reference	Module 5: Review & Rating System

Test Case ID	TC 5.4
Test Case Name	Reporting false language or misleading reviews
Preconditions	User must be logged in.
Test Steps	1. Login to the app. 2. Navigate to feature: 'Reporting false language or misleading reviews'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Reporting false language or misleading reviews works as expected • visible to all relevant users

Assignment 5 – Software Engineering (CSC-291)

Actual Result	<ul style="list-style-type: none"> • Reporting false language or misleading reviews works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-4
Module Reference	Module 5: Review & Rating System

Test Case ID	TC 5.5
Test Case Name	View uploaded reviews
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'View uploaded reviews'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • View uploaded reviews works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • View uploaded reviews works as expected • visible to all relevant users
Status	Pass
Created By	QA Team

Assignment 5 – Software Engineering (CSC-291)

Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-5
Module Reference	Module 5: Review & Rating System

Test Case ID	TC 5.6
Test Case Name	Filtering reviews
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Filtering reviews'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Filtering reviews works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Filtering reviews works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-6
Module Reference	Module 5: Review & Rating System

Test Case ID	TC 5.7
Test Case Name	Delete the Review
Preconditions	User must be logged in.
Test Steps	1. Login to the app. 2. Navigate to feature: 'Delete the Review'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Delete the Review works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Delete the Review works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-7
Module Reference	Module 5: Review & Rating System

Test Case ID	TC 5.8
Test Case Name	Send notification of reviews
Preconditions	User must be logged in.
Test Steps	1. Login to the app. 2. Navigate to feature: 'Send notification of reviews'.

	3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Send notification of reviews works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Send notification of reviews works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-8
Module Reference	Module 5: Review & Rating System

Module 6: Notification & Alert System

In this module, users and admins receive the notifications about their Agreements.

Test Case ID	TC 6.1
Test Case Name	Send Notification
Preconditions	User must be logged in.
Test Steps	1. Login to the app. 2. Navigate to feature: 'Send Notification'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Send Notification works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Send Notification works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-1
Module Reference	Module 6: Notification & Alert System

Test Case ID	TC 6.2
Test Case Name	Set Notification Preferences
Preconditions	User must be logged in.
Test Steps	1. Login to the app. 2. Navigate to feature: 'Set Notification

	Preferences'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Set Notification Preferences works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Set Notification Preferences works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-2
Module Reference	Module 6: Notification & Alert System

Test Case ID	TC 6.3
Test Case Name	View Notification History
Preconditions	User must be logged in.
Test Steps	1. Login to the app. 2. Navigate to feature: 'View Notification History'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data

Expected Result	<ul style="list-style-type: none"> • View Notification History works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • View Notification History works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-3
Module Reference	Module 6: Notification & Alert System

Test Case ID	TC 6.4
Test Case Name	Dismiss Notification
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Dismiss Notification'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Dismiss Notification works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Dismiss Notification works as expected • visible to all relevant users
Status	Pass

Assignment 5 – Software Engineering (CSC-291)

Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-4
Module Reference	Module 6: Notification & Alert System

Test Case ID	TC 6.5
Test Case Name	Customize Notification Sound
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Customize Notification Sound'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Customize Notification Sound works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Customize Notification Sound works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025

Assignment 5 – Software Engineering (CSC-291)

Test Environment	Windows 11
Use Case Reference	UC-5
Module Reference	Module 6: Notification & Alert System

Test Case ID	TC 6.6
Test Case Name	Mute Notifications
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Mute Notifications'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Mute Notifications works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Mute Notifications works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-6
Module Reference	Module 6: Notification & Alert System

Test Case ID	TC 6.7
Test Case Name	Configure Notification Types

Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Configure Notification Types'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Configure Notification Types works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Configure Notification Types works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-7
Module Reference	Module 6: Notification & Alert System

Test Case ID	TC 6.8
Test Case Name	Receive System Alerts
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Receive System Alerts'. 3. Perform the action. 4. Confirm.

Assignment 5 – Software Engineering (CSC-291)

Test Data	Sample input data
Expected Result	<ul style="list-style-type: none">• Receive System Alerts works as expected• visible to all relevant users
Actual Result	<ul style="list-style-type: none">• Receive System Alerts works as expected• visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-8
Module Reference	Module 6: Notification & Alert System

Module 7: Admin Dashboard & Analytics

In this module, admin can see all the activities taking place. This helps with the smooth running of the application.

Test Case ID	TC 7.1
Test Case Name	View User Activity Analytics
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'View User Activity Analytics'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • View User Activity Analytics works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • View User Activity Analytics works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-1
Module Reference	Module 7: Admin Dashboard & Analytics

Test Case ID	TC 7.2
Test Case Name	View Financial Analytics

Assignment 5 – Software Engineering (CSC-291)

Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'View Financial Analytics'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • View Financial Analytics works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • View Financial Analytics works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-2
Module Reference	Module 7: Admin Dashboard & Analytics

Test Case ID	TC 7.3
Test Case Name	Filter Analytics Data
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Filter Analytics Data'. 3. Perform the action. 4. Confirm.

Assignment 5 – Software Engineering (CSC-291)

Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Filter Analytics Data works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Filter Analytics Data works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-3
Module Reference	Module 7: Admin Dashboard & Analytics

Test Case ID	TC 7.4
Test Case Name	Download Analytics Report
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Download Analytics Report'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Download Analytics Report works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Download Analytics Report works as expected • visible to all relevant users

Assignment 5 – Software Engineering (CSC-291)

Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-4
Module Reference	Module 7: Admin Dashboard & Analytics

Test Case ID	TC 7.5
Test Case Name	View Admin Activities
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'View Admin Activities'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • View Admin Activities works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • View Admin Activities works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi

Assignment 5 – Software Engineering (CSC-291)

Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-5
Module Reference	Module 7: Admin Dashboard & Analytics

Test Case ID	TC 7.6
Test Case Name	Edit Admin Settings
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Edit Admin Settings'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Edit Admin Settings works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Edit Admin Settings works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-6
Module Reference	Module 7: Admin Dashboard & Analytics

Test Case ID	TC 7.7
--------------	--------

Assignment 5 – Software Engineering (CSC-291)

Test Case Name	Monitor System Health
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Monitor System Health'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Monitor System Health works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Monitor System Health works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-7
Module Reference	Module 7: Admin Dashboard & Analytics

Test Case ID	TC 7.8
Test Case Name	Generate System Performance Report
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Generate System Performance Report'.

Assignment 5 – Software Engineering (CSC-291)

	3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none">• Generate System Performance Report works as expected• visible to all relevant users
Actual Result	<ul style="list-style-type: none">• Generate System Performance Report works as expected• visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-8
Module Reference	Module 7: Admin Dashboard & Analytics

Module 8: Secure Chat & Agreement Templates

This module is used for secure chat between different users.

Test Case ID	TC 8.1
Test Case Name	Initiate Secure Chat
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Initiate Secure Chat'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Initiate Secure Chat works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Initiate Secure Chat works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-1
Module Reference	Module 8: Secure Chat & Agreement Templates

Test Case ID	TC 8.2
Test Case Name	Send Encrypted Message
Preconditions	User must be logged in.

Assignment 5 – Software Engineering (CSC-291)

Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Send Encrypted Message'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Send Encrypted Message works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Send Encrypted Message works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-2
Module Reference	Module 8: Secure Chat & Agreement Templates

Test Case ID	TC 8.3
Test Case Name	Receive Secure Message
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Receive Secure Message'. 3. Perform the action. 4. Confirm.

Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Receive Secure Message works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Receive Secure Message works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-3
Module Reference	Module 8: Secure Chat & Agreement Templates

Test Case ID	TC 8.4
Test Case Name	Terminate Chat Session
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Terminate Chat Session'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Terminate Chat Session works as expected • visible to all relevant users

Assignment 5 – Software Engineering (CSC-291)

Actual Result	<ul style="list-style-type: none"> • Terminate Chat Session works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-4
Module Reference	Module 8: Secure Chat & Agreement Templates

Test Case ID	TC 8.5
Test Case Name	Upload Agreement Template
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Upload Agreement Template'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Upload Agreement Template works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Upload Agreement Template works as expected • visible to all relevant users
Status	Pass

Assignment 5 – Software Engineering (CSC-291)

Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-5
Module Reference	Module 8: Secure Chat & Agreement Templates

Test Case ID	TC 8.6
Test Case Name	Share Agreement in Chat
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Share Agreement in Chat'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Share Agreement in Chat works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Share Agreement in Chat works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025

Assignment 5 – Software Engineering (CSC-291)

Test Environment	Windows 11
Use Case Reference	UC-6
Module Reference	Module 8: Secure Chat & Agreement Templates

Test Case ID	TC 8.7
Test Case Name	Accept Agreement
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'Accept Agreement'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • Accept Agreement works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • Accept Agreement works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-7
Module Reference	Module 8: Secure Chat & Agreement Templates

Test Case ID	TC 8.8
Test Case Name	View Agreement History
Preconditions	User must be logged in.
Test Steps	<ol style="list-style-type: none"> 1. Login to the app. 2. Navigate to feature: 'View Agreement History'. 3. Perform the action. 4. Confirm.
Test Data	Sample input data
Expected Result	<ul style="list-style-type: none"> • View Agreement History works as expected • visible to all relevant users
Actual Result	<ul style="list-style-type: none"> • View Agreement History works as expected • visible to all relevant users
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Windows 11
Use Case Reference	UC-8
Module Reference	Module 8: Secure Chat & Agreement Templates

6.2 Integration Testing:

Module 1: User Authentication

Test Case ID	TC 1.1 M01
--------------	------------

Test Case Name	User login and Authentication
Preconditions	All involved modules are functional and user is authenticated if required.
Test Steps	<ol style="list-style-type: none"> 1. Navigate to login page. 2. Enter valid username and password. 3. Click 'Login' button. 4. System authenticates credentials. 5. Redirect to dashboard. 6. Navigate to 'Profile'. 7. Click 'Edit'. 8. Update email and phone number. 9. Click 'Save'. 10. Confirm that profile info is updated.
Test Data	Username: user123 Password: pass123 Email: user@gmail.com
Expected Result	<ol style="list-style-type: none"> 1. Login successful. 2. Dashboard visible. 3. Profile updated with new info.
Actual Result	<ol style="list-style-type: none"> 1. Login successful. 2. Dashboard visible. 3. Profile changes saved.
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Web Platform Windows 11 Microsoft Edge

Module 2: Roommate Compatibility Assessment

Test Case ID	TC 2.1 M02
Test Case Name	Compatibility Form and Match Score
Preconditions	User is logged in and all matching modules are active.
Test Steps	<ol style="list-style-type: none"> 1. Navigate to compatibility form. 2. Fill out personality and lifestyle data. 3. Submit preferences. 4. System generates compatibility score. 5. Show matching profiles.
Test Data	User answers: Q1-A, Q2-B, Q3-C

Expected Result	1. Preferences saved. 2. Score calculated. 3. Matches displayed.
Actual Result	1. Data saved. 2. Match suggestions shown.
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Web App Chrome on Windows 11

Module 3: Property and Room Listings

Test Case ID	TC 3.1 M03
Test Case Name	Post and View Property Listing
Preconditions	User is logged in and has listing privileges.
Test Steps	1. Navigate to 'Post Property'. 2. Enter title, description, and upload images. 3. Submit the listing. 4. System confirms listing and redirects to detail view.
Test Data	Title: Spacious Apartment Image: 2BHK.jpg Location: New York
Expected Result	1. Listing saved. 2. Displayed in listings view.
Actual Result	1. Listed successfully. 2. Appears under 'My Listings'.
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Chrome on Windows 11

Module 4: AI-based Matching Algorithm

Test Case ID	TC 4.1 M04
Test Case Name	Generate and Display Match Scores

Preconditions	User profiles with compatibility data are present.
Test Steps	1. Submit new compatibility data. 2. AI processes input. 3. Generate scores for top matches. 4. Display top 3 profiles.
Test Data	User A input: Early riser, non-smoker
Expected Result	1. Top matches displayed with scores.
Actual Result	1. Profiles shown with correct scores.
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Web App, Edge Browser, Win 11

Module 5: Review & Rating System

Test Case ID	TC 5.1 M05
Test Case Name	Submit and View Review
Preconditions	User completed transaction with another user.
Test Steps	1. Navigate to reviews section. 2. Select transaction partner. 3. Rate with stars and comment. 4. Submit review. 5. Verify appearance on profile.
Test Data	Stars: 4 Review: Good communication.
Expected Result	1. Review saved. 2. Appears on partner profile.
Actual Result	1. Review visible immediately.
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Android App, Pixel 6

Module 6: Notification & Alert System

Test Case ID	TC 6.1 M06
--------------	------------

Test Case Name	Send Agreement Notification
Preconditions	User action triggers a system alert.
Test Steps	1. Accept rental agreement. 2. Trigger notification to other party. 3. System displays banner and logs it.
Test Data	Agreement ID: AG123456
Expected Result	1. Notification sent and logged.
Actual Result	1. Notification shown successfully.
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Mobile Web, Safari on iOS

Module 7: Admin Dashboard & Analytics

Test Case ID	TC 7.1 M07
Test Case Name	View Financial Analytics
Preconditions	Admin access granted.
Test Steps	1. Login as Admin. 2. Navigate to analytics tab. 3. Click financial reports. 4. Filter by date. 5. View charts.
Test Data	Date range: 01/01/2025 - 05/01/2025
Expected Result	1. Financial graphs display properly.
Actual Result	1. Charts render correctly.
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Admin Panel (Web)

7 Module 8: Secure Chat & Agreement Templates

Test Case ID	TC 8.1 M08
Test Case Name	Initiate Secure Chat and Share Agreement
Preconditions	Both users online and verified.

Test Steps	1. Open user chat. 2. Send greeting. 3. Click 'Attach Agreement'. 4. Upload agreement template. 5. Send file.
Test Data	File: rental_agreement.pdf
Expected Result	1. Encrypted message and file delivered.
Actual Result	1. File delivered and accessible in chat.
Status	Pass
Created By	QA Team
Date of Creation	05/30/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Secure Chat Module, Web App

System Testing:

Test Case ID	TC-SYS-ROOMIEFIND-001
Test Case Name	End-to-End Functional Verification of RoomieFind System
Preconditions	1. User has an active RoomieFind account. 2. All modules (Authentication, Compatibility Assessment, Property Listings, AI Matching, Chat, Notifications, Dashboard) are deployed and accessible. 3. User is using a supported browser or mobile app with active internet connection.
Test Steps	1. Navigate to RoomieFind login page. 2. Enter valid credentials and log in. 3. Complete roommate compatibility questionnaire. 4. View compatibility score and suggested roommates. 5. Post a property listing with photos, price, and location. 6. Search and filter listings using budget and location preferences. 7. Use AI-based suggestions to find top matched roommates. 8. Initiate a secure chat with one of the

	<p>suggested matches.</p> <p>9. Upload and share a digital rental agreement via chat.</p> <p>10. Accept agreement and view shared history.</p> <p>11. Receive system notifications for chat, agreement, and profile match.</p> <p>12. Update profile settings and delete account.</p>
Test Data	<p>Username: roomie_user</p> <p>Password: Match@2025</p> <p>Questionnaire: Answers to lifestyle, habits, preferences</p> <p>Listing: 2BHK, Rent: \$700, Location: Downtown</p> <p>Search Filters: Budget: \$600–\$800, Location: Downtown</p> <p>Chat Message: Hello! Let's discuss this listing.</p> <p>Agreement File: rental_terms.pdf</p> <p>Preferences: Notifications enabled, Theme: Light</p> <p>Profile Change: Email: user@roomie.com, Status: Seeking Roommate</p>
Expected Result	<p>1. User logged in and redirected to dashboard.</p> <p>2. Compatibility form submitted and scored.</p> <p>3. Roommate suggestions based on compatibility displayed.</p> <p>4. Property listing posted and visible publicly.</p> <p>5. Filtered search results returned correctly.</p> <p>6. AI-generated roommate matches shown.</p> <p>7. Chat initiated with selected user.</p> <p>8. Agreement file sent successfully.</p> <p>9. Agreement accepted and logged.</p> <p>10. Notifications received for major actions.</p> <p>11. Profile updated and account deleted successfully.</p>
Actual Result	<p>1. Login successful.</p> <p>2. Questionnaire saved and score shown.</p>

	3. Matches visible on dashboard. 4. Listing submitted and confirmed. 5. Filter working correctly. 6. Top matches listed with compatibility %. 7. Chat message sent. 8. File uploaded and delivered in chat. 9. Agreement accepted. 10. Notifications logged and delivered. 11. Profile deleted and redirected to login.
Status	Pass
Created By	QA Team
Date of Creation	05/31/2025
Executed By	Zeeshan Qureshi
Date of Execution	05/31/2025
Test Environment	Cross-platform Testing: Windows 11 Chrome, Android App, iOS App

8 Acceptance Test Cases – RoomieFind

This document outlines the acceptance test cases for the RoomieFind platform, ensuring that the implemented system aligns with the specified functional requirements and is ready for academic submission.

Test Case ID	Feature	Test Description	Input	Expected Output	Pass Criteria
ATC-01	User Authentication	Verify user can register and log in with valid credentials	Valid email, username, password	User account created; successful login redirects to dashboard	User is logged in and redirected
ATC-02	Personality Quiz	Ensure user can complete and submit the compatibility quiz	Responses to all quiz questions	Quiz is submitted and stored; AI receives quiz data	Quiz completion message appears
ATC-03	AI-Based Matching	Validate that the system generates roommate matches based on quiz results	User profile with quiz data	List of compatible users displayed based on AI logic	Minimum 3 matches shown, relevance verified

Assignment 5 – Software Engineering (CSC-291)

ATC-04	Room & Property Listings	Check that verified room listings are displayed and searchable	Filter by location, rent range, room type	Filtered room listings displayed with key details	Results match filters
ATC-05	Secure Chat & Agreements	Confirm users can message matched roommates and send/view agreement templates	Message + selected agreement template	Message sent; agreement downloadable/viewable	Message and agreement are visible to both users
ATC-06	Review & Rating System	Test users can submit a review and rating for a roommate or property	Rating (1–5 stars) + written review	Review saved and displayed publicly on profile/listing	Review appears with correct content
ATC-07	Notifications & Alerts	Verify users receive real-time alerts for matches, messages, and system updates	Trigger events (new match, message)	Notification pop-up and inbox update	Notification appears within 5 seconds
ATC-08	Admin Dashboard	Check admin access to user stats and system metrics	Admin login	Dashboard with charts, logs, and analytics	Metrics load and display correctly

9 2.X Acceptance Testing Strategy – RoomieFind

The acceptance testing plan for RoomieFind is designed in line with standard software engineering concepts taught in CSC291. It ensures that the system is validated from the perspective of contractual compliance, end-user experience, administrator readiness, and real-world usability.

9.1 Types of Acceptance Testing

Type	Purpose	Application in RoomieFind
Contract Acceptance Testing	To verify that the software meets all the requirements defined in the SRS and agreed-upon project scope.	All modules (e.g., AI matching, chat, listings, admin dashboard) will be tested against documented requirements.
User Acceptance Testing (UAT)	To validate that the software meets the needs of users. Conducted by student testers or peers.	Students will test room quiz, matching, chat, and reviews to ensure the system is usable and intuitive.
Operational Acceptance Testing	To confirm the system is operationally ready for deployment by admin or IT staff.	Admin users will test dashboard, system metrics, content moderation tools, and notification configurations.
Field Testing	Real-world feedback through Alpha (internal) and Beta (external) environments.	Alpha: Team and classmates simulate use cases in dev environment. Beta: Optional limited release to external users for feedback.

9.2 Mapping with Generic Testing Types

Generic Testing Type	RoomieFind Application
Functional Testing	Ensures features like registration, matching, chat, listings, and notifications work as expected.
Non-Functional Testing	Focuses on usability (senior-friendly design), performance (response time of AI match and chat), and compatibility.
Structural Testing	Will be done during development – ensuring internal structures (e.g., method calls, flow of control) are tested.

Testing Related to Changes / Regression Testing	Applied when fixing bugs or updating features—ensures no existing functionality is broken after changes.
-------------------------------------------------	----------------------------------------------------------------------------------------------------------

10 Test Deliverables

Deliverable	Description
1. Test Case Specifications	Detailed test case documents outlining each test's ID, name, preconditions, steps, data, expected and actual results, status, environment, etc. (e.g., Unit, Integration, System test cases).
2. Test Data	All required input data, configurations, and mock datasets used to execute the test cases. This may include login credentials, patient records, forum entries, emergency request data, etc.
3. Test Logs / Execution Logs	Step-by-step records of test execution, including time stamps, testers involved, systems used, and any anomalies noted. Useful for traceability and audit.
S4. Test Summary Report	An executive-level report summarizing overall test activities and results. Includes test coverage, number of test cases executed, pass/fail rates, and any outstanding issues.
5. Test Incident Reports (Defect Reports)	Documented reports of defects/bugs discovered during testing, including severity, status, environment, and screenshots/logs if available.
6. Test Environment Setup Document	A detailed description of the test environment(s) used: OS, browser versions, mobile platforms, server configuration, databases, etc.
7. Traceability Matrix	A mapping between requirements and test cases to ensure full test coverage.
8. Test Tools (if applicable)	Description and configuration details of tools used for automation, bug tracking (e.g., Selenium, JIRA), test management (e.g., TestRail), or monitoring.
9. Final Test Closure Report (Optional but ideal)	A summary to close out the testing phase, highlighting lessons learned, open issues, and overall quality assessment.

6. Conclusion

The testing phase plays a vital role in validating the functionality, reliability, and performance of the RoomieFind platform, ensuring it meets all requirements outlined in the Software Requirements Specification (SRS). Through careful planning, detailed test case design, and

thorough execution, each module—from user authentication and personality quizzes to AI-based matching, secure chat, and admin dashboard—has been verified to function correctly. Test deliverables, including test cases, execution logs, incident reports, and summary documents, provide clear insight into the quality assurance process. By aligning testing activities closely with the overall project timeline, RoomieFind is positioned for a successful launch, enhancing the experience of users seeking compatible roommates and trusted housing options.

7. Appendices

Name	Description	Possible States / Values	Related Operations	Requirements
LoginAttempts	Maximum allowed login attempts before lockout	3	Login Validation	Security – Lockout after multiple failures
UserStatus	Tracks user login state	LoggedIn, LoggedOut, Suspended	Login, Logout, Session Timeout	Authentication
CompatibilityScore	Match score generated by AI	0–100	Calculate Score, Update Feedback	Roommate Compatibility
QuestionnaireResponses	User lifestyle/preference answers	Multiple Choice/Rating Scale	Submit Responses	Compatibility Matching
PropertyListing	Rental details posted by user	Draft, Posted, Expired	Create, Edit, Delete	Listing Management
SearchFilter	Filters for finding properties or roommates	Location, Budget, Gender	Apply Filter, Reset	Search Functionality
AgreementTemplate	Legal template used for rental agreement	PDF/DOC File	Upload, Share, Accept	Secure Agreement Management
ChatMessage	Encrypted communication between users	Text, Image, File	Send, Receive, Delete	Secure Chat Module
Notification	Informs user of system actions	Email, In-App, SMS	Trigger Notification	Alert & Notification System

UserProfile	Personal data associated with account	Active, Inactive	Update Profile, Delete Profile	User Management
LoginAttempts	Maximum allowed login attempts before logout	3	Login Validation	Security – Lockout after multiple failures

8. References

- **EEE 829 - Standard for Software and System Test Documentation**
<https://ieeexplore.ieee.org/document/4118471>
- **ISTQB Glossary of Testing Terms**
<https://glossary.istqb.org/en/search>

9. Plagiarism Report:

Turnitin Originality Report <small>Processed on 2-June-2025 15:10 PKT ID: 2637938350 Word Count: 11019 Submitted: 1</small>					
Report By Zeeshan Qureshi.	<table> <tr> <th>Similarity Index</th><th>Similarity by Source</th></tr> <tr> <td>6%</td><td> Internet Sources: 4% Publications: 0% Student Papers: 2% </td></tr> </table>	Similarity Index	Similarity by Source	6%	Internet Sources: 4% Publications: 0% Student Papers: 2%
Similarity Index	Similarity by Source				
6%	Internet Sources: 4% Publications: 0% Student Papers: 2%				
1% match (student papers from 30-Sep-2018) Submitted to Higher Education Commission Pakistan on 2018-09-30					
1% match (Internet from 15-Mar-2014) http://www.mpd-so.com/pdfs/beijing/architect/huang.pdf					
< 1% match (student papers from 02-Oct-2018) Submitted to Higher Education Commission Pakistan on 2018-10-02					
< 1% match (Internet from 20-Apr-2023) https://www.coursehero.com/file/p2vjh6s0/2-Face-recognition-is-not-available-3High-Speed-Internet-Required-11-19/					
< 1% match (Internet from 17-Dec-2024) https://www.coursehero.com/file/86908615/Proposal-templatedocx/					
< 1% match (Internet from 31-Oct-2024) https://academic-master.com/an-android-application-for-autonomous-file-sharing-indelay-tolerant-network/					
<p>Project Category: ? A-Desktop Application/Information System ? B-Web Application ? C-Problem Solving and Artificial Intelligence ? D-Simulation and Modeling ? E-Smartphone Application ? F-Smartphone Game ? G-Networks ? H-Image Processing ?</p> <p>Other Abstract Elder Care is a platform which helps senior citizens to make simple connections to caregivers, volunteers, and supporting them to stay healthier and independent. Most aged persons today face problems like isolation, restricted health care access, and insufficient prompt assistance, yet available solutions continue to be ineffective. This project will fill these gaps by providing an user friendly platform that guarantees strong connections between seniors citizens and caregiving services. One of the major problems with the existing system is the lack of a specialized digital solution that targets the specific needs of older adults. With Elder Care, seniors will be able to communicate more effectively with caregivers, response times will be faster, and care plans can be changed to specific requirements. Real-time monitoring and emergency response capabilities will also increase security, while a network of volunteers will offer added support. With a well-designed and accessible solution, this platform will be crucial in enhancing the independence, security, and well-being of aged people. 1. Introduction Elder Care platform is a digital solution designed to meet the needs of senior citizens by connecting them with caregivers, volunteers and basic services. This platform aims to finish or at least reduce the gap between the senior citizens and the services they need. The families are getting busier and many elderly individuals face difficulties like isolation and access to healthcare. The Elder Care platform will serve as a centralized hub allowing senior citizens to connect with caregivers and volunteers. The system is user friendly which will help senior citizens to connect to caregivers easily. There are many caregiving services available but it is difficult for senior citizens to find them. <u>The purpose of this document is to present a detailed plan for the development and implementation of Elder Care platform.</u> This platform will change the lives of many senior citizens. This platform will give them a better way of living and independence. The platform also aims to provide security to senior citizens in case of critical and emergency situations. 2. Problem Statement The population of elder people is growing rapidly in the world and with this rise comes rise in concern</p>					