Communities A Social Media Platform(SM02)
Test Plan

Harshit Pant CS21BTECH11021   Satpute Aniket Tukaram CS21BTECH11056

Mahin Bansal CS21BTECH11034   Burra Vishal Mathews CS21BTECH11010

Table of Contents

Test Overview

Communities is an online social media platform that allows users to create and join communities based on their interests. Users can create communities, post content, comment on posts, and interact with other users. The platform also includes other features such as chat, search, and recommendations and certain level of moderation.

This document outlines the test plan for Communities, including the modules to be tested, the types of tests to be conducted, and the test analysis.

The test plan includes the following operations/modules that will be tested:

- Authentication Modules
  - Login Module
  - Signup Module
  - Token Authentication Module
  - E2EE Module
- User Modules
  - Guest User Module
  - Registered User Module
  - Admin User Module
  - Moderator User Module
  - Superuser Module
- System Modules
  - Cache Manager Module
  - Recommender System Module
  - Job Scheduler Module
  - Services Module
    * Feed Service
    * Comments Service
    * Post Service

- ∗ Search Service
- ∗ Votes Service
- ∗ Chat Service
- ∗ Connections Service
- ∗ Scoring Service
- DB Access Modules
  - User Module
  - Post Module
  - Comment Module
  - Vote Module
  - Chat Module
  - Community Module
- UI Modules
  - Listing Module
- Other Modules
  - Notification Module
  - Reporting Module
  - Moderation Module

The test plan includes the following sections:

- Test Overview: A brief overview of the test plan which also lists the operations/modules that will be tested.
- Test Details: Detailed information about the unit tests, integration testing, system testing, and performance testing.
- Test Analysis: A summary of the test statistics, functional test report, and performance test report.
  - Test Statistics: The number of classes, methods, modules tested, test cases, and test cases failed.
  - Functional Test Report: A summary of the functional test cases for each module, including the type of testing method used.
  - Performance Test Report: A summary of the performance test plan and report.

Test Details

```
Integrating testing
    The order in which you will integrate your modules and test cases for integrated modules.
System testing
Performance testing
```

Unit Tests

Authentication Module

Table 1: Authentication Module Unit Test

S.No	Module Name	Function	Conditions to be tested	Inputs	Expected Output	Status
UT-1.1.a	Login Module	Login user using Interanl sign-in	Incorrect Credentials	user: unknown-username **or** wrong-password, DB-status: user-not-found **or** wrong-password (from User Module)	return: invalid-credentials	F

Table 1: Authentication Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Inputs	Expected Output	Status
UT-1.1.b	Login Module	Login User using Google OAuth	Incorrect Credentials (OAuth)	user: incorrect-email **or** wrong-password, OAuth-status: user-not-registered **or** wrong-password (from OAuth)	return: invalid-credentials	F
UT-1.1.c	Login Module	Login User	Correct Credentials	user: registered-username **and** correct-password, DB-status: OK (from User Module or OAuth)	return: OK **and** JSON Web-token	P
UT-1.2.a	Token Module	Validate JWT	Invalid Token	cookie: invalid-token	The module extracts the header and the payload from the request and recalculates the signature and matches it with the token, in this case it does not match and hence returns **returns:** invalid-token	F
UT-1.3.a	Signup Module	Signup User	Username already taken	user: already-taken-username **and** password **and** email-id, DB-status: username-taken	return: username-taken	F
UT-1.3.b	Signup Module	Signup User	Unverified email id	user: username **and** password **and** unverfied-email-id, email-service-status: email-verification-timeout	return: email-unverified	F
UT-1.3.c	Signup Module	Signup User	Passwords Don't Match	user: username **and** invalid-password **and** email-id	return: password-mismatch	F
UT-1.3.d	Signup Module	Signup User	Valid Credentials	user: username **and** password **and** email-id, DB-status: user-created	return: user-created **and** JWT and redirects user to home page	P
UT-1.3.a	E2EE Module	TO-BE-FILLED	Invalid Key			F

User Module

S.No	Module Name	Conditions to be tested	Test Data	Expected Output	Status
UT-2.1.a					

System Module

S.No	Module Name	Conditions to be tested	Test Data	Expected Output	Status
UT-3.1.a					

Database Access Module

User Record Module

Table 4: User Record Module Unit Test

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.1.a	User Module	getUser	Invalid Username	user: invalid-user-name	return: invalid-user-name	F
UT-4.1.b	User Module	getUser	No User by this Username	user: unknown-user, DB-status: no-such-user	return: no-such-user	F
UT-4.1.c	User Module	getUser	Exisiting username	user: username, DB-status: OK	return: OK **and** user-record	P
UT-4.1.d	User Module	updateUser	Invalid Username	user: invalid-user-name	return: invalid-user-name	F
UT-4.1.e	User Module	updateUser	No User by this Username	user: username, DB-status: OK	return: OK	F
UT-4.1.f	User Module	updateUser	Edit User Settings (User setting is a 6-tuple) - Invalid Tuple - Invalid Profile Visibility	user: new-user-record (invalid integer for profile visibility)	return: invalid-profile-visibility	F
UT-4.1.g	User Module	updateUser	Edit User Settings (User setting is a 6-tuple) - Invalid Tuple - Invalid Chat Request Setting	user: new-user-record (invalid integer for chat request)	return: invalid-chat-request-setting	F
UT-4.1.h	User Module	updateUser	Edit User Settings (User setting is a 6-tuple) - Invalid Tuple - Invalid Notification Setting - remaining 4 values	user: new-user-record (invalid 4-tuple for notification)	return: invalid-notification-setting	F
UT-4.1.i	User Module	updateUser	Edit User Settings (User setting is a 6-tuple) - Valid Tuple	user: username	return: OK	P

Table 4: User Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.1.j	User Module	updateUser	Edit email-id - Invalid email-id	user: new-user-record (email-id: invalid)	return: invalid-email	F
UT-4.1.k	User Module	updateUser	Edit email-id - Valid email-id	user: valid-new-user-record	return: OK	P
UT-4.1.l	User Module	updateUser	Edit user bio	user: username, DB-status: OK	return: OK	P
UT-4.1.m	User Module	deleteUser	Invalid Username	user: invalid-user, DB-status: user-not-found	return: invalid-user-name	F
UT-4.1.n	User Module	deleteUser	No User by this Username	user: username, DB-status: no-such-user	return: no-such-user	F
UT-4.1.o	User Module	deleteUser	Exisiting username	user: username, DB-status: OK	return: user-deleted	P

Post Record Module

Table 5: Post Record Module Unit Test

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.2.a	Post Module	get Post	Invalid PostID/Deleted PostID	post: unknown-post-id, DB-status: post-id-not-found	return: post-not-found	F
UT-4.2.b	Post Module	get Post By Community	Invalid CommunityID/ Deleted Community	community: unknown-community-id **and** sort: sort-by **and** filter: filter-by, DB-status: community-not-found	return: community-not-found	F
UT-4.2.c	Post Module	get Post By Community	Invalid Sort-By	community: community-id **and** sort: invalid-sort-by **and** filter: filter-by	return: invalid-sort	F

Table 5: Post Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.2.d	Post Module	get Post By Community	Invalid Filter-by	community: community-id **and** sort: sort-by **and** filter: invalid-filter-by	return: invalid-filter-by	F
UT-4.2.e	Post Module	get Post By Community	Valid Request	community: community-id **and** sort: sort-by **and** filter: filter-by, DB-status: OK	return: OK **and** post-list	F
UT-4.2.f	Post Module	get Post By User	Invalid UserID	post: unknown-user-id **and** sort: sort-by **and** filer: filter-by, DB-status: user-not-found	return: user-not-found	F
UT-4.2.g	Post Module	get Post By User	Valid Request	post: user-id **and** sort: sort-by **and** filer: filter-by, DB-status: OK	return: OK **and** post-list	P
UT-4.2.h	Post Module	get Post By User	Invalid Sort-By	post: user-id **and** sort: invalid-sort-by **and** filer: filter-by	return: invalid-sort	F
UT-4.2.i	Post Module	get Post By User	Invalid Filter-by	post: user-id **and** sort: sort-by **and** filer: invalid-filter-by	return: invalid-filter-by	F
UT-4.2.j	Post Module	get Post By Score User	Invalid UserID (Used for user-feed generation)	post: unknown-user-id, DB-status: user-not-found	return: user-not-found	F
UT-4.2.k	Post Module	get Post By Score Trending	Valid Request	no-input	return: post-list	P
UT-4.2.l	Post Module	get Post By Tag	Invalid TagID	tag: unknown-tag-id **and** sort: sort-by **and** filter: filter-by, DB-status: tag-not-found	return: tag-not-found	F

Table 5: Post Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.2.m	Post Module	get Post By Tag	Invalid Sort-By	tag: tag-id **and** sort: invalid-sort-by **and** filter: filter-by	return: invalid-sort	F
UT-4.2.n	Post Module	get Post By Tag	Invalid Filter-By	tag: tag-id **and** sort: sort-by **and** filter: invalid-filter-by	return: invalid-filter-by	F
UT-4.2.o	Post Module	get Post By Tag	Valid Request	tag: tag-id **and** sort: sort-by **and** filter: filter-by, DB-status: OK	return: OK **and** post-list	P
UT-4.2.p	Post Module	get Post By Query String	Invalid Sort-By	query-string: search-string **and** sort: invalid-sort-by **and** filter: filter-by	return: invalid-sort-by	F
UT-4.2.q	Post Module	get Post By Query String	Invalid Filter-By	query-string: search-string **and** sort: sort-by **and** filter: invalid-filter-by	return: invalid-filter-by	F
UT-4.2.r	Post Module	get Post By Query String	Valid Request	query-string: search-string **and** sort: sort-by **and** filter: filter-by, DB-status: OK	return: OK **and** post-list	P
UT-4.2.s	Post Module	update Post	Invalid PostID	post: unknown-post-id, DB-status: post-not-found	return: post-not-found	F
UT-4.2.t	Post Module	delete Post	Invalid PostID	post: post-id, DB-status: post-not-found	return: post-not-found	F
UT-4.2.u	Post Module	delete Post By User	invalid UserID	post: user-id, DB-status: user-not-found	return: user-not-found	F
UT-4.2.v	Post Module	delete Post By User	valid UserID	post: user-id, DB-status: OK	return: OK	P
UT-4.2.w	Post Module	delete Post By Community	invalid CommunityID	post: community-id, DB-status : community-not-found	return: community-not-found	F

Table 5: Post Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.2.x	Post Module	delete Post By Community	valid CommunityID	post: community-id, DB-status: OK	return: OK	P

Comment Record Module

Table 6: Comment Record Module Unit Test

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.3.a	Comment Module	get Comment	Invalid CommentID	comment: unknown-comment-id, DB-status: comment-not-found	return: comment-not-found	F
UT-4.3.b	Comment Module	get Comment	Valid Comment	comment: known-comment, DB-status: OK	return: OK **and** comment	P
UT-4.3.c	Comment Module	get Comment By Post	Invalid PostID	post: unknown-post-id, DB-status: post-not-found	return: post-not-found	F
UT-4.3.d	Comment Module	get Comment By Parent	Invalid Parent CommentID	comment: unknown-comment-id, DB-status: comment-not-found	return: comment-not-found	F
UT-4.3.e	Comment Module	get Comment By Parent	Valid Comment	comment: known-comment, status: OK	return: OK **and** comment	P
UT-4.3.f	Comment Module	get Comments By Ancestor	Invalid Ancestor CommentID	comment: unknown-comment-id, DB-status: comment-not-found	return: comment-not-found	F
UT-4.3.g	Comment Module	get Comments By Ancestor	Valid Ancestor CommentID	comment: known-comment, status: OK	return: OK **and** comment	P

Table 6: Comment Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.3.h	Comment Module	update Comment	Invalid CommentID	comment: unknown-comment-id, DB-status: comment-not-found	return: comment-not-found	F
UT-4.3.i	Comment Module	update Comment	Valid Comment	comment: known-comment, status: OK	return: OK	P
UT-4.3.j	Comment Module	delete Comment	Invalid CommentID	comment: unknown-comment-id, DB-status: comment-not-found	return: comment-not-found	F
UT-4.3.k	Comment Module	delete Comment	Valid Comment	comment: known-comment, status: OK	return: OK	P
UT-4.3.l	Comment Module	delete Comment By Post	Invalid PostID	post: unknown-post-id, DB-status: post-not-found	return: post-not-found	F
UT-4.3.m	Comment Module	delete Comment By Post	Valid PostID	post: known-post-id, DB-status: OK	return: OK	P

Vote Record Module

Table 7: Vote Record Module Unit Test

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.4.a	Vote Module	get Vote	Invalid VoteID	vote: unknown-vote-id, DB-status: vote-not-found	return: vote-not-found	F
UT-4.4.b	Vote Module	get Vote	Valid Vote	vote: known-vote, DB-status: OK	return: OK **and** vote	P
UT-4.4.c	Vote Module	get Vote By Post	Invalid PostID	post: unknown-post-id, DB-status: post-not-found	return: post-not-found	F

Table 7: Vote Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.4.d	Vote Module	get Vote-count By Post	Invalid PostID	post: unknown-post-id, DB-status: post-not-found	return: vote-not-found	F
UT-4.4.e	Vote Module	get Vote-count By Post	Valid PostID	post: known-post-id, DB-status: OK	return: OK **and** vote-count	P
UT-4.4.f	Vote Module	get Vote By Post	Valid PostID	post: known-post-id, DB-status: OK	return: OK **and** vote-list	P
UT-4.4.g	Vote Module	get Vote By Post-User	Invalid PostID	post: unknown-post-id **and** user: user-id, DB-status: post-not-found	return: post-not-found	F
UT-4.4.h	Vote Module	get Vote By Post-User	Invalid UserID	post: post-id **and** user: unknown-user-id, DB-status: user-not-found	return: user-not-found	F
UT-4.4.i	Vote Module	get Vote By Post-User	Valid Request	post: post-id **and** user: user-id, DB-status: OK	return: OK **and** vote	P
UT-4.4.j	Vote Module	update Vote	Invalid VoteID	vote: unknown-vote-id, DB-status: vote-not-found	return: vote-not-found	F
UT-4.4.k	Vote Module	update Vote	Valid Vote	vote: known-vote, DB-status: OK	return: OK	P
UT-4.4.l	Vote Module	delete Vote	Invalid VoteID	vote: unknown-vote-id, DB-status: vote-not-found	return: vote-not-found	F
UT-4.4.m	Vote Module	delete Vote	Valid Vote	vote: known-vote, DB-status: OK	return: OK	P
UT-4.4.n	Vote Module	delete Vote By Post	Invalid PostID	post: unknown-post-id, DB-status: post-not-found	return: post-not-found	F
UT-4.4.o	Vote Module	delete Vote By Post	Valid PostID	post: known-post-id, DB-status: OK	return: OK	P

Table 7: Vote Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.4.p	Vote Module	delete Vote By Post-User	Invalid PostID	post: unknown-post-id **and** user: user-id, DB-status: post-not-found	return: post-not-found	F
UT-4.4.q	Vote Module	delete Vote By Post-User	Invalid UserID	post: post-id **and** user: unknown-user-id, DB-status: user-not-found	return: user-not-found	F
UT-4.4.r	Vote Module	update Vote By Post-User	Invalid PostID	post: unknown-post-id **and** user: user-id, DB-status: post-not-found	return: post-not-found	F
UT-4.4.s	Vote Module	update Vote By Post-User	Invalid UserID	post: post-id **and** user: unknown-user-id, DB-status: user-not-found	return: user-not-found	F
UT-4.4.t	Vote Module	update Vote By Post-User	Valid Request	post: post-id **and** user: user-id, DB-status: OK	return: OK	P

Chat(Private Messages) Record Module

Table 8: Chat Record Module Unit Test

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.5.a	Chat Module	get Chat	Invalid ChatID	chat: unknown-chat-id, DB-status: chat-not-found	return: chat-not-found	F
UT-4.5.b	Chat Module	get Chat	Valid Chat	chat: known-chat, DB-status: OK	return: OK	P
UT-4.5.c	Chat Module	delete Chat	Invalid ChatID	chat: unknown-chat-id, DB-status: chat-not-found	return: chat-not-found	F
UT-4.5.d	Chat Module	delete Chat	Valid Chat	chat: known-chat, DB-status: OK	return: OK	P

Table 8: Chat Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.5.e	Chat Module	delete Message-Chat	Invalid MessageID	message: unknown-message-id **and** chat: chat-id, DB-status: message-not-found	return: message-not-found	F
UT-4.5.f	Chat Module	delete Message-Chat	Invalid ChatID	message: message-id **and** chat: invalid-chat-id, DB-status: OK	return: invalid-chat-id	F
UT-4.5.f	Chat Module	delete Message-Chat	Valid Message	message: message-id **and** chat: chat-id, DB-status: OK	return: OK	P

Message(Group and Private) Record Module

Table 9: Message Record Module Unit Test

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.6.a	Message Module	get Message	Invalid MessageID	message: unknown-message, DB-status: message-not-found	return: message-not-found	F
UT-4.6.b	Message Module	get Message	Valid Message	message: known-message, DB-status: OK	return: OK **and** message	P
UT-4.6.c	Message Module	get Messages By Chat	Invalid ChatID	chat: unknown-chat-id, DB-status: chat-not-found	return: chat-not-found	F
UT-4.6.d	Message Module	get Messages By Chat	Valid Chat	chat: known-chat-id, DB-status: OK	return: OK **and** message-list	P
UT-4.6.e	Message Module	get Messages By Group	Invalid GroupID	group: unknown-group-id, DB-status: group-not-found	return: group-not-found	F
UT-4.6.f	Message Module	get Messages By Group	Valid Group	group: known-group-id, DB-status: OK	return: OK **and** message-list	P

Table 9: Message Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.6.g	Message Module	update Message	Invalid MessageID	message: unknown-message, DB-status: message-not-found	return: message-not-found	F
UT-4.6.h	Message Module	update Message	Valid Message	message: known-message, DB-status: OK	return: OK	P
UT-4.6.i	Message Module	delete Message	Invalid MessageID	message: unknown-message, DB-status: message-not-found	return: message-not-found	F
UT-4.6.j	Message Module	delete Message	Valid Message	message: known-message, DB-status: OK	return: OK	P
UT-4.6.k	Message Module	delete Messages By Chat	Invalid ChatID	chat: unknown-chat, DB-status: chat-not-found	return: chat-not-found	F
UT-4.6.l	Message Module	delete Messages By Chat	Valid Chat	chat: known-chat, DB-status: OK	return: OK	P
UT-4.6.m	Message Module	delete Messages By Group	Invalid GroupID	group: unknown-group, DB-status: group-not-found	return: group-not-found	F
UT-4.6.n	Message Module	delete Messages By Group	Valid Group	group: known-group, DB-status: OK	return: OK	P

Group Record Module

Table 10: Group Record Module Unit Test

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.7.a	Group Module	get Group	Invalid GroupID	group: unknown-group, DB-status: group-not-found	return: group-not-found	F
UT-4.7.b	Group Module	get Group	Valid Group	group: known-group, DB-status: OK	return: OK **and** group	P

Table 10: Group Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.7.c	Group Module	delete Group	Invalid GroupID	group: unknown-group, DB-status: group-not-found	return: group-not-found	F
UT-4.7.d	Group Module	delete Group	Valid Group	group: known-group, DB-status: OK	return: OK	P

User__chat Record Module

Table 11: User-Chat Record Module Unit Test

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.8.a	User-Chat Module	get User and Chat	Invalid UserID	user: unknown-user **and** chat-id, DB-status: user-not-found	return: user-not-found	F
UT-4.8.b	User-Chat Module	get User and Chat	Invalid ChatID	user: user-id **and** chat: unknown-chat, DB-status: chat-not-found	return: chat-not-found	F
UT-4.8.c	User-Chat Module	get User and Chat	Invalid UserID and ChatID tuple	user: known-user, chat: known-chat, DB-status: unknown-user-chat-tuple	return: user-not-in-chat	F
UT-4.8.d	User-Chat Module	get User and Chat	Valid Request	user: known-user, chat: known-chat, status: OK	return: OK	P
UT-4.8.e	User-Chat Module	delete User and Chat	Invalid UserID	user: unknown-user **and** chat-id, DB-status: user-not-found	return: user-not-found	F
UT-4.8.f	User-Chat Module	delete User and Chat	Invalid ChatID	user: user-id **and** chat: unknown-chat, DB-status: chat-not-found	return: chat-not-found	F

Table 11: User-Chat Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.8.g	User-Chat Module	delete User and Chat	Invalid UserID and ChatID tuple	user: known-user, chat: known-chat, DB-status: unknown-user-chat-tuple	return: user-not-in-chat	F
UT-4.8.h	User-Chat Module	delete User and Chat	Valid Request	user: known-user, chat: known-chat, status: OK	return: OK	P

User_group Record Module

Table 12: User-Group Record Module Unit Test

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.9.a	User-Group Module	get User and Group	Invalid UserID	user: unknown-user **and** group-id, DB-status: user-not-found	return: user-not-found	F
UT-4.9.b	User-Group Module	get User and Group	Invalid GroupID	user: user-id **and** group: unknown-group, DB-status: group-not-found	return: group-not-found	F
UT-4.9.c	User-Group Module	get User and Group	Invalid UserID and GroupID tuple	user: known-user, group: known-group, DB-status: unknown-user-group-tuple	return: user-not-in-group	F
UT-4.9.d	User-Group Module	get User and Group	Valid Request	user: known-user, group: known-group, status: OK	return: OK	P
UT-4.9.e	User-Group Module	delete User and Group	Invalid UserID	user: unknown-user **and** group-id, DB-status: user-not-found	return: user-not-found	F

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.9.f	User-Group Module	delete User and Group	Invalid GroupID	user: user-id **and** group: unknown-group, DB-status: group-not-found	return: group-not-found	F
UT-4.9.g	User-Group Module	delete User and Group	Invalid UserID and GroupID tuple	user: known-user, group: known-group, DB-status: unknown-user-group-tuple	return: user-not-in-group	F
UT-4.9.h	User-Group Module	delete User and Group	Valid Request	user: known-user, group: known-group, status: OK	return: OK	P

Community Record Module

Table 13: Community Record Module Unit Test

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.10.a	Community Module	get Community	Invalid CommunityID	community: unknown-community, DB-status: community-not-found	return: community-not-found	F
UT-4.10.b	Community Module	get Community	Valid Community	community: known-community, DB-status: OK	return: OK	P
UT-4.10.c	Community Module	delete Community	Invalid CommunityID	community: unknown-community, DB-status: community-not-found	return: community-not-found	F
UT-4.10.d	Community Module	delete Community	Valid Community	community: known-community, DB-status: OK	return: OK	P

Table 13: Community Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.10.e	Community Module	update Community Bio	Invalid CommunityID	community: unknown-community **and** bio: community-bio, DB-status: community-not-found	return: community-not-found	F
UT-4.10.f	Community Module	update Community Bio	Valid Community	community: known-community **and** bio: community-bio, DB-status: OK	return: OK	P
UT-4.10.g	Community Module	update Community Settings	Invalid CommunityID	community: unknown-community **and** settings: setting-tuple, DB-status: community-not-found	return: community-not-found	F
UT-4.10.h	Community Module	update Community Settings	Invalid SettingTuple (Community type, post type, member privileges)	community: known-community **and** settings: invalid-setting-tuple	return: Invalid-setting-tuple	F
UT-4.10.i	Community Module	update Community Settings	Valid Community	community: known-community **and** settings: setting-tuple, DB-status: OK	return: OK	P
UT-4.10.j	Community Module	update Community Image	Invalid CommunityID	community: unknown-community **and** image: community-image, DB-status: community-not-found	return: community-not-found	F
UT-4.10.k	Community Module	update Community Image	Valid Community	community: known-community **and** image: community-image, DB-status: OK	return: OK	P

Joined Community Record Module

Table 14: Joined Community Record Module Unit Test

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.11.a	Joined Community Module	get User-Community	Invalid UserID	user: unknown-user **and** community: community-id, DB-status: user-not-found	return: user-not-found	F
UT-4.11.b	Joined Community Module	get User-Community	Invalid CommunityID	user: user-id **and** community: unknown-community, DB-status: community-not-found	return: community-not-found	F
UT-4.11.c	Joined Community Module	get User-Community	Invalid UserID and CommunityID tuple	user: known-user, community: known-community, DB-status: unknown-user-community-tuple	return: user-not-in-community	F
UT-4.11.d	Joined Community Module	get User-Community	Valid Request	user: known-user, community: known-community, status: OK	return: OK	P
UT-4.11.e	Joined Community Module	get User-Community Status (requested, invited, joined, blocked)	Invalid UserID	user: unknown-user **and** community: community-id, DB-status: user-not-found	return: user-not-found	F
UT-4.11.f	Joined Community Module	get User-Community Status (requested, invited, joined, blocked)	Invalid CommunityID	user: user-id **and** community: unknown-community, DB-status: community-not-found	return: community-not-found	F
UT-4.11.g	Joined Community Module	get User-Community Status (requested, invited, joined)	Invalid UserID and CommunityID tuple	user: known-user, community: known-community, DB-status: unknown-user-community-tuple	return: user-not-in-community	F

Table 14: Joined Community Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.11.h	Joined Community Module	get User-Community Status (requested, invited, joined, blocked)	Valid Request	user: known-user, community: known-community, status: OK	return: OK **and** user-status	P
UT-4.11.i	Joined Community Module	delete User-Community	Invalid UserID	user: unknown-user **and** community: community-id, DB-status: user-not-found	return: user-not-found	F
UT-4.11.j	Joined Community Module	delete User-Community	Invalid CommunityID	user: user-id **and** community: unknown-community, DB-status: community-not-found	return: community-not-found	F
UT-4.11.k	Joined Community Module	delete User-Community	Invalid UserID and CommunityID tuple	user: known-user, community: known-community, DB-status: unknown-user-community-tuple	return: user-not-in-community	F
UT-4.11.l	Joined Community Module	delete User-Community	Valid Request	user: known-user, community: known-community, status: OK	return: OK	P
UT-4.11.m	Joined Community Module	update User-Community Status (requested, invited, joined, blocked)	Invalid UserID	user: unknown-user **and** community: community-id, DB-status: user-not-found	return: user-not-found	F
UT-4.11.n	Joined Community Module	update User-Community Status (requested, invited, joined, blocked)	Invalid CommunityID	user: user-id **and** community: unknown-community, DB-status: community-not-found	return: community-not-found	F

Continued on next page

Table 14: Joined Community Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.11.o	Joined Community Module	update User-Community Status (requested, invited, joined, blocked)	Invalid UserID and CommunityID tuple	user: known-user, community: known-community, DB-status: unknown-user-community-tuple	return: user-not-in-community	F
UT-4.11.p	Joined Community Module	update User-Community Status (requested, invited, joined, blocked)	Valid Request	user: known-user, community: known-community, status: (joined / invited/ requested)	return: OK	P
UT-4.11.q	Joined Community Module	update User-Community Status (requested, invited, joined, blocked)	Invalid Status	user: known-user, community: known-community, status: invalid-status	return: invalid-status	F
UT-4.11.r	Joined Community Module	update User-Community privileges	Invalid UserID	user: unknown-user **and** community: community-id, DB-status: user-not-found	return: user-not-found	F
UT-4.11.s	Joined Community Module	update User-Community privileges	Invalid CommunityID	user: user-id **and** community: unknown-community, DB-status: community-not-found	return: community-not-found	F
UT-4.11.t	Joined Community Module	update User-Community privileges	Invalid UserID and CommunityID tuple	user: known-user, community: known-community, DB-status: unknown-user-community-tuple	return: user-not-in-community	F
UT-4.11.u	Joined Community Module	update User-Community privileges	Invalid Privileges	user: known-user, community: known-community, privileges: invalid-privileges	return: invalid-privileges	F

Table 14: Joined Community Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.11.v	Joined Community Module	update User-Community privileges	Valid Request	user: known-user, community: known-community, privileges: valid-privileges	return: OK	P
UT-4.11.w	Joined Community Module	update User-Community privileges	Valid Request	user: known-user, community: known-community, privileges: valid-privileges	return: OK	P
UT-4.11.x	Joined Community Module	update User-Community privileges	Valid Request	user: known-user, community: known-community, privileges: valid-privileges	return: OK	P

Blocked User Record Module

Table 15: Blocked User Record Module Unit Test

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.12.a	Blocked Module	get Blocked Users(1) By UserID(2)	Invalid UserID	user: unknown-user, DB-status: user-not-found	return: user-not-found	F
UT-4.12.b	Blocked Module	get Blocked Users(1) By UserID(2)	Valid Request	user: known-user, DB-status: OK	return: OK **and** blocked-list	P
UT-4.12.c	Blocked Module	delete Blocked User(1) By UserID(2)	Invalid UserID	user1: unknown-user **and** user2: user-id, DB-status: user-not-found	return: user-not-found	F
UT-4.12.d	Blocked Module	delete Blocked User(1) By UserID(2)	Invalid UserID	user1: user-id **and** user2: unknown-user, DB-status: user-not-found	return: user-not-found	F
UT-4.12.e	Blocked Module	delete Blocked User(1) By UserID(2)	Tuple Not Present	user1: user-id **and** user2: known-user, DB-status: tuple-not-found	return: tuple-not-found	F

Table 15: Blocked User Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.12.f	Blocked Module	delete Blocked User(1) By UserID(2)	Valid Request	user1: known-user **and** user2: known-user, DB-status: OK	return: OK	P

Reported User Record Module

Table 16: Reported User Record Module Unit Test

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.13.a	Reported Module	get By UserID	Invalid UserID	user: unknown-user, DB-status: user-not-found	return: user-not-found	F
UT-4.13.b	Reported Module	get By UserID	Valid Request	user: known-user, DB-status: OK	return: OK **and** report-list	P
UT-4.13.c	Reported Module	get By ReportID	Invalid ReportID	report: unknown-report, DB-status: report-not-found	return: report-not-found	F
UT-4.13.d	Reported Module	get By ReportID	Valid Request	report: known-report, DB-status: OK	return: OK **and** report	P
UT-4.13.e	Reported Module	delete By UserID	Invalid UserID	user: unknown-user, DB-status: user-not-found	return: user-not-found	F
UT-4.13.f	Reported Module	delete By UserID	Valid Request	user: known-user, DB-status: OK	return: OK	P
UT-4.13.g	Reported Module	delete By ReportID	Invalid ReportID	report: unknown-report, DB-status: report-not-found	return: report-not-found	F
UT-4.13.h	Reported Module	delete By ReportID	Valid Request	report: known-report, DB-status: OK	return: OK	P

Table 16: Reported User Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.13.i	Reported Module	update Report Status By ReportID	Invalid ReportID	report: unknown-report, status: report-status, DB-status: report-not-found	return: report-not-found	F
UT-4.13.j	Reported Module	update Report Status By ReportID	Invalid Report Status	report: known-report, status: invalid-status, DB-status: OK	return: invalid-status	F
UT-4.13.k	Reported Module	update Report Status By ReportID	Valid Request	report: known-report, status: report-status, DB-status: OK	return: OK	P

Roles Record Module

Table 17: Roles Record Module Unit Test

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.14.a	Roles Module	get Role By User-Community	Invalid UserID	user: unknown-user, community: community-id	return: user-not-found	F
UT-4.14.b	Roles Module	get Role By User-Community	Invalid CommunityID	user: user-id, community: unknown-community	return: community-not-found	F
UT-4.14.c	Roles Module	get Role By User-Community	Invalid UserID and CommunityID tuple	user: known-user, community: known-community, DB-status: unknown-user-community-tuple	return: user-not-in-community	F
UT-4.14.d	Roles Module	get Role By User-Community	Valid Request	user: known-user, community: known-community, DB-status: OK	return: OK **and** role	P

Table 17: Roles Record Module Unit Test (Continued)

S.No	Module Name	Function	Conditions to be tested	Test Data	Expected Output	Status
UT-4.14.e	Roles Module	update Role By User-Community	Invalid UserID	user: unknown-user, community: community-id	return: user-not-found	F
UT-4.14.f	Roles Module	update Role By User-Community	Invalid CommunityID	user: user-id, community: unknown-community	return: community-not-found	F
UT-4.14.g	Roles Module	update Role By User-Community	Invalid UserID and CommunityID tuple	user: known-user, community: known-community, DB-status: unknown-user-community-tuple	return: user-not-in-community	F
UT-4.14.h	Roles Module	update Role By User-Community	Invalid Role	user: known-user, community: known-community, role: invalid-role	return: invalid-role	F
UT-4.14.i	Roles Module	update Role By User-Community	Valid Request	user: known-user, community: known-community, role: valid-role	return: OK	P
UT-4.14.j	Roles Module	delete Role By User-Community	Invalid UserID	user: unknown-user, community: community-id	return: user-not-found	F
UT-4.14.k	Roles Module	delete Role By User-Community	Invalid CommunityID	user: user-id, community: unknown-community	return: community-not-found	F
UT-4.14.l	Roles Module	delete Role By User-Community	Invalid UserID and CommunityID tuple	user: known-user, community: known-community, DB-status: unknown-user-community-tuple	return: user-not-in-community	F
UT-4.14.m	Roles Module	delete Role By User-Community	Valid Request	user: known-user, community: known-community, DB-status: OK	return: OK	P

UI Module

- One of the UI Module that will be tested is the Listing Service Module which is a module which is present at the client side of the application. This module is responsible for listing the communities, posts, comments and users. The module will be tested for the following functionalities:

- List Community (Search Results and Feed)
- List Posts (Search Results and Community and Feed)
- List Comments
- List Users (Search Results and Feed)
- List Reports (Admin, Moderator, Superuser(platform related), User(All reports on a User))
- List Community Requests (Admin, Moderator)
- List Notifications
- List User-Community (Admin, Moderator)
- List Followers
- List Following
- List Blocked Users (Admin, Moderator, User(One User bloked by other), Superuser(blocked-from-platform))

Table 18: Listing Service Module Unit Test

S.No	Module Name	Functionality	Conditions to be tested	Test Data	Expected Output	Status
UT-5.1.a	Listing Service Module	List Community	Some Community Record in the List of Records is Invalid	list: list of community records in JSON	return: invalid-list	F
UT-5.1.b	Listing Service Module	List Community	Valid List of Community Records	list: list of community records in JSON	return: returns HTML	P
UT-5.1.c	Listing Service Module	List Posts	Some Post Record in the List of Records is Invalid	list: list of post records in JSON	return: invalid-list	F
UT-5.1.d	Listing Service Module	List Posts	Valid List of Post Records	list: list of post records in JSON	return: returns HTML	P
UT-5.1.e	Listing Service Module	List Comments	Some Comment Record in the List of Records is Invalid	list: list of comment-tree records in JSON	return: invalid-list	F
UT-5.1.f	Listing Service Module	List Comments	Valid List of Comment Records	list: list of comment-tree records in JSON	return: returns HTML	P
UT-5.1.g	Listing Service Module	List Users	Some User Record in the List of Records is Invalid	list: list of user records in JSON	return: invalid-list	F
UT-5.1.h	Listing Service Module	List Users	Valid List of User Records	list: list of user records in JSON	return: returns HTML	P
UT-5.1.i	Listing Service Module	List Reports	Some Report Record in the List of Records is Invalid	list: list of report records in JSON	return: invalid-list	F
UT-5.1.j	Listing Service Module	List Reports	Valid List of Report Records	list: list of report records in JSON	return: returns HTML	P

Table 18: Listing Service Module Unit Test (Continued)

S.No	Module Name	Functionality	Conditions to be tested	Test Data	Expected Output	Status
UT-5.1.k	Listing Service Module	List Requests	Some Request Record in the List of Records is Invalid	list: list of request records in JSON	return: invalid-list	F
UT-5.1.l	Listing Service Module	List Requests	Valid List of Request Records	list: list of request records in JSON	return: returns HTML	P
UT-5.1.m	Listing Service Module	List Notifications	Some Notification Record in the List of Records is Invalid	list: list of notification records in JSON	return: invalid-list	F
UT-5.1.n	Listing Service Module	List Notifications	Valid List of Notification Records	list: list of notification records in JSON	return: returns HTML	P
UT-5.1.o	Listing Service Module	List User-Community	Some User-Community Record in the List of Records is Invalid	list: list of user-community records in JSON	return: invalid-list	F
UT-5.1.p	Listing Service Module	List User-Community	Valid List of User-Community Records	list: list of user-community records in JSON	return: returns HTML	P
UT-5.1.q	Listing Service Module	List Followers	Some Follower Record in the List of Records is Invalid	list: list of follower records in JSON	return: invalid-list	F
UT-5.1.r	Listing Service Module	List Followers	Valid List of Follower Records	list: list of follower records in JSON	return: returns HTML	P
UT-5.1.s	Listing Service Module	List Following	Some Following Record in the List of Records is Invalid	list: list of following records in JSON	return: invalid-list	F
UT-5.1.t	Listing Service Module	List Following	Valid List of Following Records	list: list of following records in JSON	return: returns HTML	P
UT-5.1.u	Listing Service Module	List Blocked Users	Some Blocked User Record in the List of Records is Invalid	list: list of blocked user records in JSON	return: invalid-list	F
UT-5.1.v	Listing Service Module	List Blocked Users	Valid List of Blocked User Records	list: list of blocked user records in JSON	return: returns HTML	P

Integration Testing

```
Integrating different modules and testing
    Unit tested submodules combined and tested
    Use bottom-up approach for integration of modules
```

S.No	Modules Integrated	Condition to be tested	Test Data	Expected Output	Status
1.1					

System Testing

Testing of system as a whole

Performance Testing

Performance testing plan

Test Analysis

We discussed the following seven types of black box testing in the class: equivalence class partitioning, bou

Performance test plan/report

Test Statistics

- Number of classes:
- Number of methods:
- Number of modules tested:
- Number of test cases:
- Number of test cases failed:

Functional Test Report

Functional test summary in the form of a table
    Test case count for each module
    Type of testing method used
        equivalence class partitioning
        boundary value analysis
        cause-effect graphing
        pair-wise testing
        special cases
        error guessing
        state based testing

S.No	Module Name	Test Case Count	Testing Method

Performance Test Report