



IT-314: Software Engineering

Group - 14

VibeHub

System Testing

INDEX

Black Box Testing Documentation.....	2
Project title: Social media platform.....	2
Testing Type: Black Box Testing/.....	2
1. Introduction.....	2
2. Test Plan.....	3
3. Test Cases.....	4
Sign up test.....	4
Login test.....	7
Send Message.....	9
Follow/unfollow user.....	11
Like/dislike post.....	12
Search user.....	13
Create Post.....	14
Edit profile.....	16
Notification.....	17
Delete post.....	19
4. Acceptance Testing.....	20

Black Box Testing Documentation

Project title: Social media platform

Testing Type: Black Box Testing

Purpose: To verify the website's functionality aligns with requirements and provides a seamless user experience.

1. Introduction

Objective:

The objective of black box testing is to evaluate the functionality and usability of the website from an end-user perspective, without concerning the internal code structure.

Scope: This testing focuses on major functionalities, including user authentication, image posting, profile viewing, commenting, and notification features.

Assumptions:

1. Users have access to a stable internet.
2. Users will interact primarily through standard web browsers (e.g., Chrome, Firefox).

Testing Tools Used:

- Browser-based manual testing
- Automation tools such as Selenium

2. Test Plan

Functional Areas to be Tested:

1. User Authentication

- Signup
- Login/Logout
- Password Reset

2. User Profile Management

- Profile Viewing and Editing
- Privacy Settings

3. Content Interaction

- Posting Images
- Liking and Commenting
- Viewing Feeds
- Bookmarking post

4. Notifications

- Comment and Like Notifications
- Friend/Follower Requests

3. Test Cases

Sign up test

1) Username:

Valid:

- A first name with less than 15 characters, following standard naming conventions.
- A valid first name adhering to common naming convention.
- A valid first name within the standard length limit.

Invalid:

- Invalid due to special characters.
- Invalid due to being too long.

2) Email:

Valid:

- A valid email address adhering to standard formatting.
- A valid email address complying with common conventions.
- A valid email address facilitating effective communication.

Invalid:

- Invalid due to a missing domain.
- Invalid due to containing numeric characters in the domain.
- Invalid due to a missing top-level domain.

3) Password:

Valid:

- A strong password containing a combination of uppercase letters, lowercase letters, numbers, and special characters.

- A password with more than 8 characters, not commonly used, and meets security standards.

Invalid:

- Invalid if it does not meet the specified criteria for a secure password.
- Invalid if it is too simple.
- Invalid if it lacks a combination of uppercase letters, lowercase letters, numbers, or special characters.

Equivalence Class partitioning:

Valid Equivalence Classes:

- Username : any non-empty string which is valid.
- Email: any valid non-empty string which is valid.
- Password: any valid input which is valid.

Invalid Equivalence Classes:

- Username : any non-empty string which is invalid.
- Email: any valid non-empty string which is invalid.
- Password: any valid input which is invalid.

Boundary Value Analysis:

Lower Boundaries:

- Username, email: Empty string
- Password: Any string with less than 8 characters.

Upper Boundaries:

- Username: string of size greater than 14.

Tests:

Username	Email	Password	Expected o/p	Actual o/p	reason
""	""	""	Invalid	Invalid	All fields are empty.
"Aadi"	"Aadi@example"	"pass"	Invalid	Invalid	Username too short, email missing domain and top-level domain, password too weak.
"AadiyaPithadiya"	"aaditya@gmail.com"	"Aadi@2005!"	Valid	Valid	All fields meet the required criteria.
"Kavan1234"	"kavan@gmail"	"pass1234"	Invalid	Invalid	Email missing top-level domain, password too simple.
"AadityaPithadiyaaaaaaaa"	"long@domain.com"	"lonG!"	Invalid for username, valid for email and password	Invalid	Username too long.
"AadiyaPithadiya"	"aaditya@gmail.com"	"123456789"	Invalid for password	Invalid	password lacks complexity.
"Apache-%16"	"firstname@abc.com"	"Complex@Password2"	Invalid for username	Invalid	Username contains special

					characters.
"Exam"	"aadi@gmail.com"	"SuperSecure#2025"	Valid	Valid	All fields satisfy requirement s.
"parth&*"	"parth@site-with-number s123.com"	"Password1!"	Invalid for username and email	Invalid	Username has special characters, invalid email.

Login test

1) Email:

Valid:

- Email should already be registered.

Invalid:

- Email not registered.

2) Password:

Valid:

- Should match the original password.

Invalid:

- Invalid if it does not match the original password.

Equivalence Class partitioning:

Valid Equivalence Classes:

- Email, password: Any valid combination which is already used in sign up.

Invalid Equivalence Classes:

- Email, password: Combination not used in sign up.

Boundary Value Analysis:

Lower Boundaries:

- email, password: Empty string

Email	Password	Expected Output	Actual Output	Reason
202201224@daiict.ac.in	Aadi@2005	Valid	Valid	Both email and password meet validity criteria.
""	P@ssword	Invalid	Invalid	Empty email field
kavan	P@ssw0rd!	Invalid	Invalid	Missing domain and '@' character in the email address.
random@123	P@ssw0rd!	Invalid	Invalid	Invalid domain.
test@example.com	PASS1234	Invalid	Invalid	Invalid email or password.
notregistered@example.com	P@ssw0rd!	Invalid	Invalid	Email address not registered in the system.

Send Message

1) Message Content:

Valid:

- A non-empty message within the character limit.

Invalid:

- An empty message.

2) Recipient User:

Valid:

- A valid recipient user whom the user is following.

Invalid:

- An invalid user.

Equivalence Class Partitioning:

Valid Equivalence Class:

- Message content: any non-empty string.
- Recipient: A valid user ID that exists.

Invalid Equivalence Class:

- Message content: Empty message.
- Recipient: non-existing or not followed user.

Boundary Value Analysis:

Lower Bound:

- Empty message content.

Message content	Recipient username	Expected Outcome	Actual Outcome	Reason
“Hello!”	ValidUser	Message sent	Message Sent	All fields are valid
“”	valid	Error	Error	Message content is empty, cannot send an empty message
“How are You!!”	User123	Error	Error	Recipient does not exist in the system.
“Good Morning”	NotAfollower	error	error	Sender is not following the recipient.

Follow/unfollow user

1) Action Type:

Follow:- User clicks the follow button to follow the other user.

Unfollow:- User clicks the unfollow button to unfollow the other user.

2) Recipient User:

Valid:- A user that exists in the system.

Invalid:- An invalid user that is not on the platform.

Equivalence Class Partitioning:

Valid Equivalence Class:

- Action type: follow/unfollow is a valid option.
- Recipient: a valid user ID that exists in the system.

Invalid Equivalence Class:

- Recipient: A user ID that doesn't exist.

Action	Recipient Username	Expected Output	Actual Output	Reason
Follow	validUser	Following	Following	Valid follow request to an eligible user.
Unfollow	valid_user	Unfollow Successful	Success	Valid unfollow request for a followed user.
Follow	self_user	No action	No action	Cannot follow oneself.
Unfollow	self_user	No action	No action	Cannot unfollow oneself.

Like/dislike post

1) Action Type:

Like:- Users like a post that they have not already liked.

Dislike:- Users dislike a post that they have already liked.

2) Post ID:

Valid:- A post that exists in the system.

Invalid:- A post that does not exist or is deleted.

Equivalence Class Partitioning:

Valid Equivalence Class:

- Action type: like/dislike is a valid option.
- Post should be existing in the database.

Invalid Equivalence Class:

- Action type: No actions taken.

Action	Post ID	Expected Output	Actual Output	Reason
Like	12345	Heart should be filled	Success	Valid like action on an existing post.
Dislike	12345	Heart should be unfilled	Success	Valid dislike action on a liked post.

Search user

1) Search Query:

Valid:- A non-empty string that matches an existing username or part of a username.

Invalid:- An empty search string or a query that does not match any existing usernames.

2) Search Results:

Valid: One or more users that match the search query.

Equivalence Class Partitioning:

Valid Equivalence Class:

- Search query: any non-empty string that partially or fully matches existing usernames.
- Search results: one or more valid users returned as matches.

Invalid Equivalence Class:

- Search query: empty or non-matching string.
- Search results: no matching users.

3) Boundary Value Analysis:

Lower Bound:

- Empty search query.

Search Query	Expected Output	Actual Output	Reason
"John"	Matches Found	Success	Valid search query; returns matching users.
"john_doe"	Exact Match Found	Success	Exact username match found in the system.
""	No users listed	No users listed	Empty search query; search action not taken.
"non_existent_user"	No Matches Found	No users listed	No users match the query, returning no results.

Create Post

1) Caption:

Valid:- Any valid non-empty string.

Invalid:- Empty string.

2) Add From computer:

Input type: Image

Valid:- Any existing image.

Invalid:- Empty file.

Equivalence Class partitioning:

Valid Equivalence Classes:

- Caption : any valid non-empty string.

- image/video: any valid input.

Invalid Equivalence Classes:

- image/video: empty file.

Boundary Value Analysis:

Lower Boundaries:

- image/video: empty file.

Caption	Image/Video File	Expected Output	Actual Output	Reason
"A beautiful sunset"	valid_image.jpg	Valid	Valid	Both caption and image file are valid non-empty inputs.
""	valid_image.jpg	Valid	Valid	You can Post Without caption also
!"	valid_video.mp4	Valid	Valid	Caption is a non-empty string,
"A beautiful sunset"	invalid_image	Invalid	Invalid	The image file path or input is incorrect or does not exist, making it invalid.

Edit profile

1) Bio:

Valid:- A non-empty bio within the character limit.

Invalid:- An empty bio.

2) Gender:

Valid:- A valid gender selection (e.g., Male, Female).

Invalid:- An unselected or unspecified gender.

3) Profile Photo:

Valid:- A valid image file (e.g., .jpg, .png, .jpeg) that exists.

Invalid:- An empty file or an unsupported file format.

Equivalence Class Partitioning:

Valid Equivalence Classes:

- Bio,Gender,Profile Photo: Any one valid input or change.

Invalid Equivalence Classes:

- None of the input fields changed.

Boundary Value Analysis:

Lower Bound: Empty bio, unspecified gender, and empty file for profile photo.

Bio	Gender	Profile Photo	Expected Output	Actual Output	Reason
"Hello, I love photography!"	Male	valid_photo.jpg	Profile Updated	Success	All fields meet validity criteria.
""	Male	valid_photo.jpg	Bio not updated.	Bio not updated.	Bio is empty, update not displayed.
"Travel enthusiast"	Female	""	No change	No change	Profile photo not selected.
"Outdoor lover"	Female	large_photo.jpeg	Profile Updated	Success	All fields meet the requirements.

Notification

1) Notification Type:

Like Notification:

- User receives a notification when someone likes their post.

2) Notification Display:

Valid:

- Notifications are displayed in the notifications symbol when triggered by an action.

Invalid:

- No notification is displayed due to a system error, blocked user, or duplicate notification.

Equivalence Class Partitioning:

Valid Equivalence Classes:

- Notification Type: valid actions.
- Notification Display: notifications appear in the panel as expected.

Invalid Equivalence Classes:

- Notification Type: unsupported or duplicate notifications.
- Notification Display: no notification is displayed due to an error or blockage.

Boundary Value Analysis:

Lower Bound:

- No notifications present.

Upper Bound:

- Multiple notifications displayed for the same event, exceeding the maximum number allowed.

Notification Type	Triggered By	Expected Output	Actual Output	Reason
Like Notification	like on post	Notification	Notification Received	User receive a notification when their post is liked by another user.
Like Notification	like on post	No Notification	No Notification	No notification generated when a user likes their own post.
Like Notification	duplicate like	No Notification	No Notification	Duplicate like does not trigger a new notification.

Delete post

1) Action Type:

Delete post:- Delete a selected post made by the user.

2) Post ID:

Valid:- A post that exists in the system.

Invalid:- A post that does not exist or is deleted.

3) Equivalence Class Partitioning:

Valid Equivalence Class:- Action type: delete is a valid option.

Invalid Equivalence Class:- Action type: No actions taken.

Action	Post ID	Expected Output	Actual Output	Reason
Delete post	12345	Post should be deleted	Post deleted	Valid action

4. Acceptance Testing

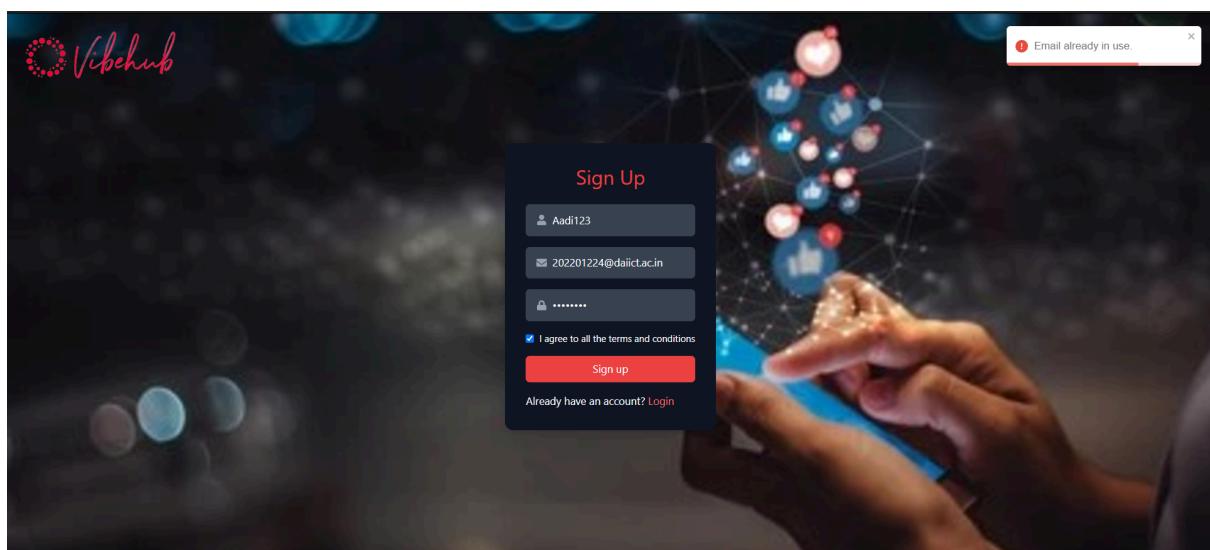
1. sign up:

Username: Aadi123

Email id : 202201224@daiict.ac.in

Password : Abc123**

Output : Email already in use

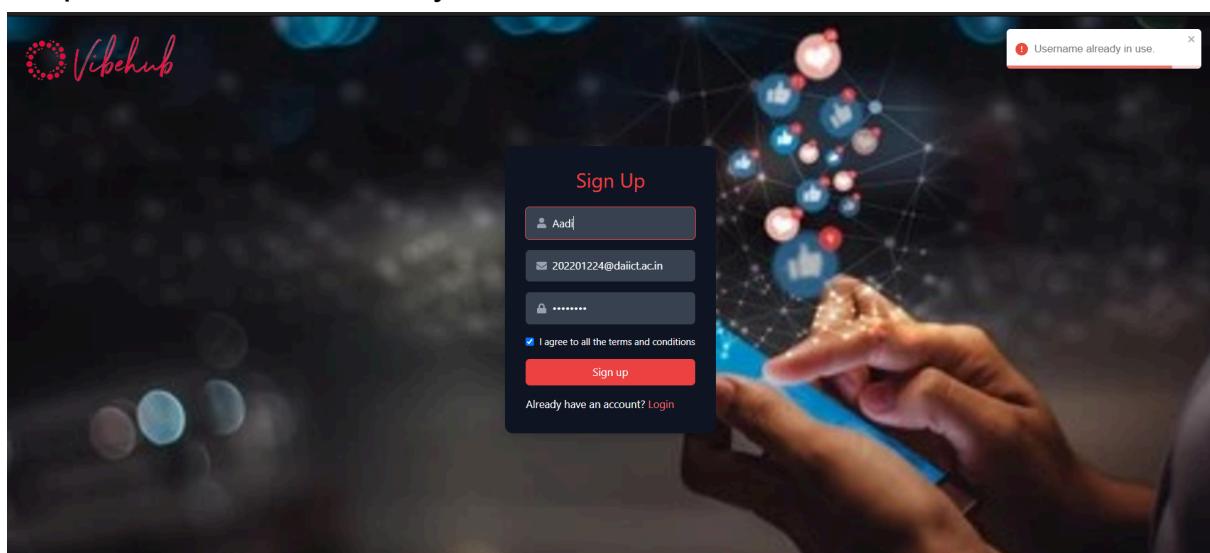


Username: Aadi

Email id : aaditya123@gmail.com

Password : Abc123**

Output : Username already in use

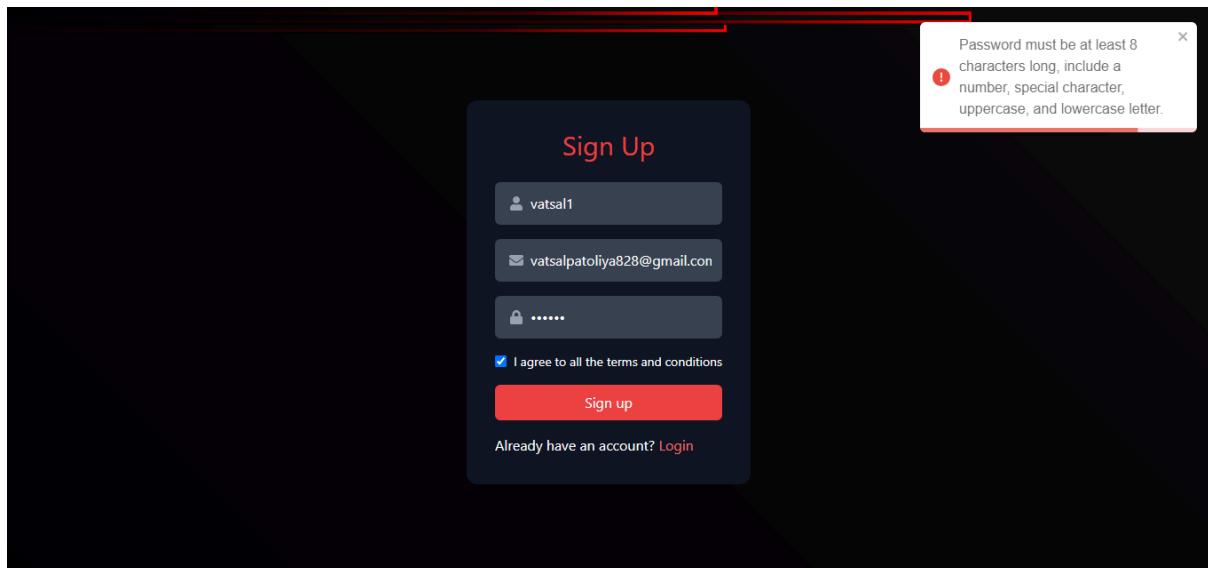


Username: vatsal1

Email id : vatsalpatoliya828@gmail.com

Password :vatsal

Output : Password must be at least 8 characters long, include a number, special character, uppercase, and lowercase letter.

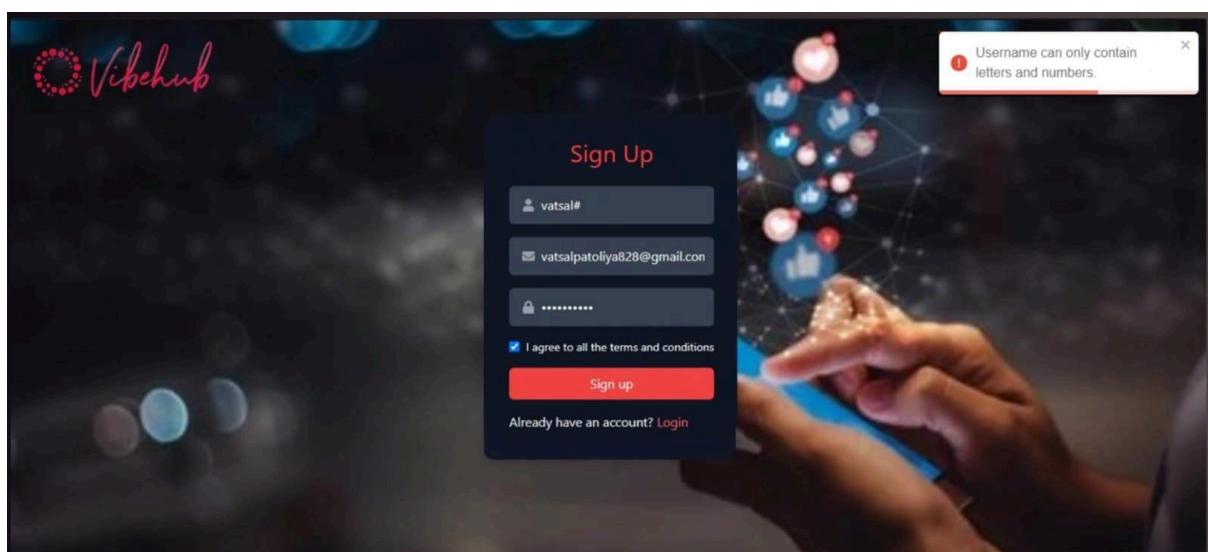


Username: vatsal#

Email id : vatsalpatoliya828@gmail.com

Password :Vatsal001#

Output :Username can only contain letters and numbers.

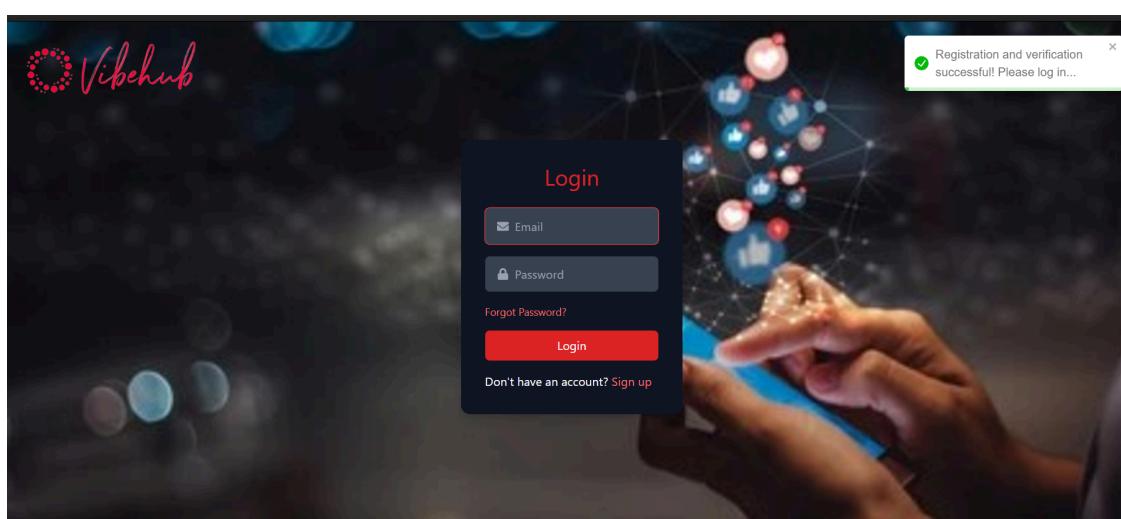
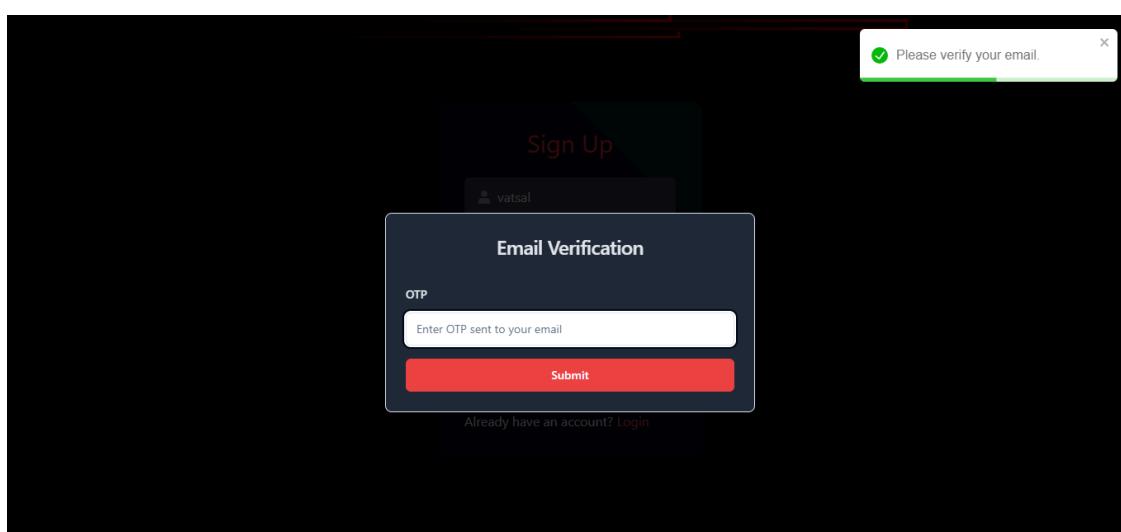
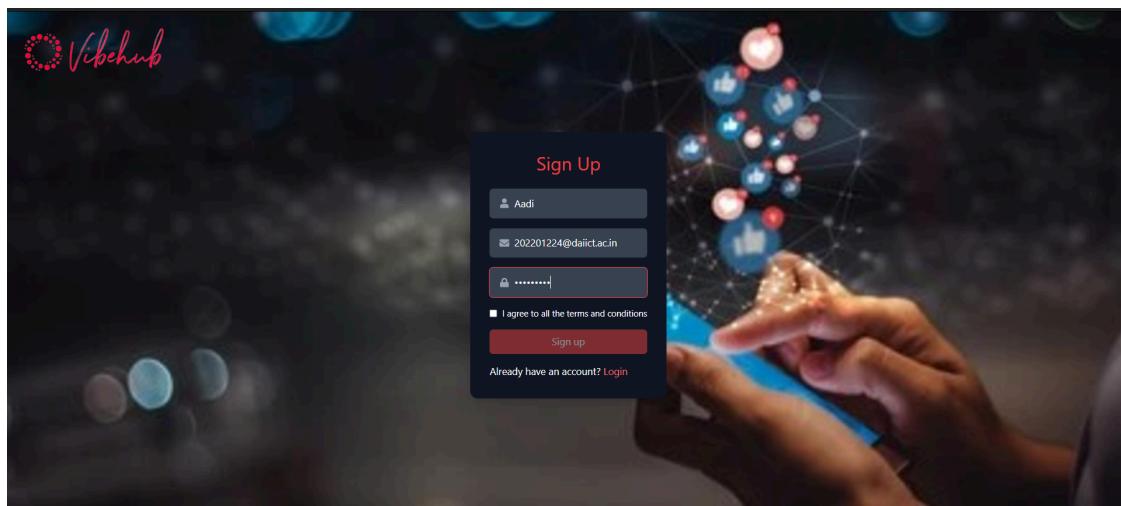


Username: Aadi

Email id : 202201224@daiict.ac.in

Password :Abc123**abc

Output : valid

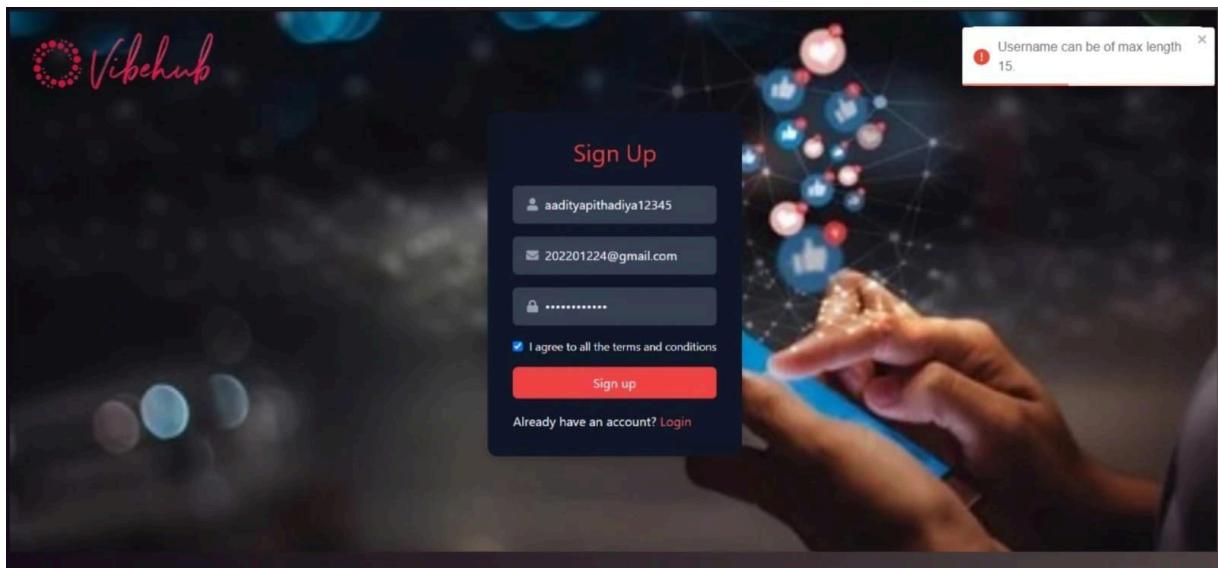


Username: aadityapithadiya123415

Email id :202201224@gmail.com

Password :Abc123**

Output :invalid

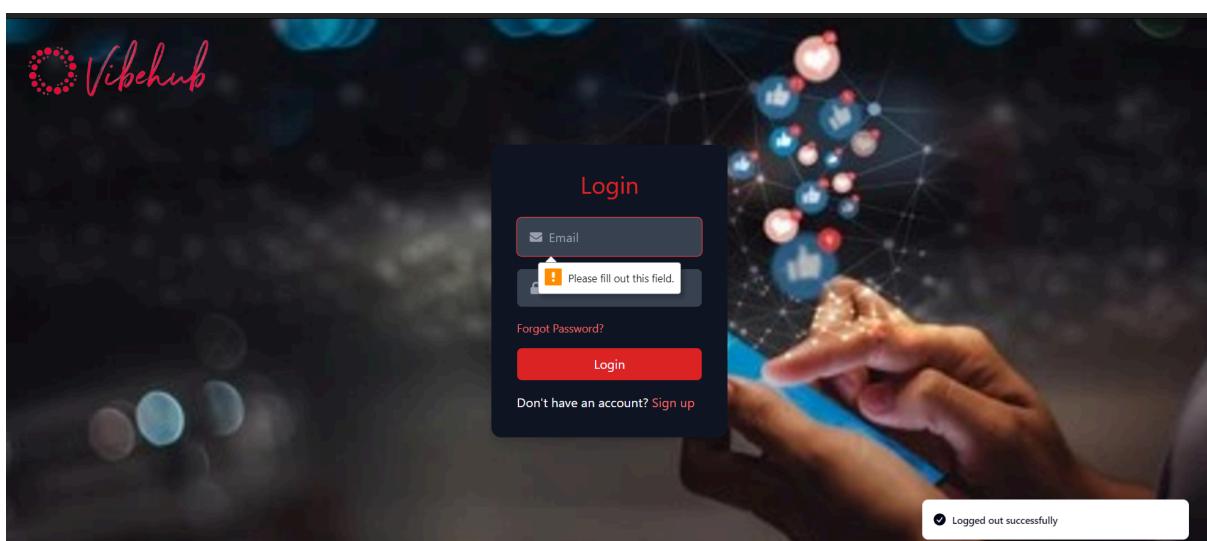


2. Login

Email id : “”

Password : “”

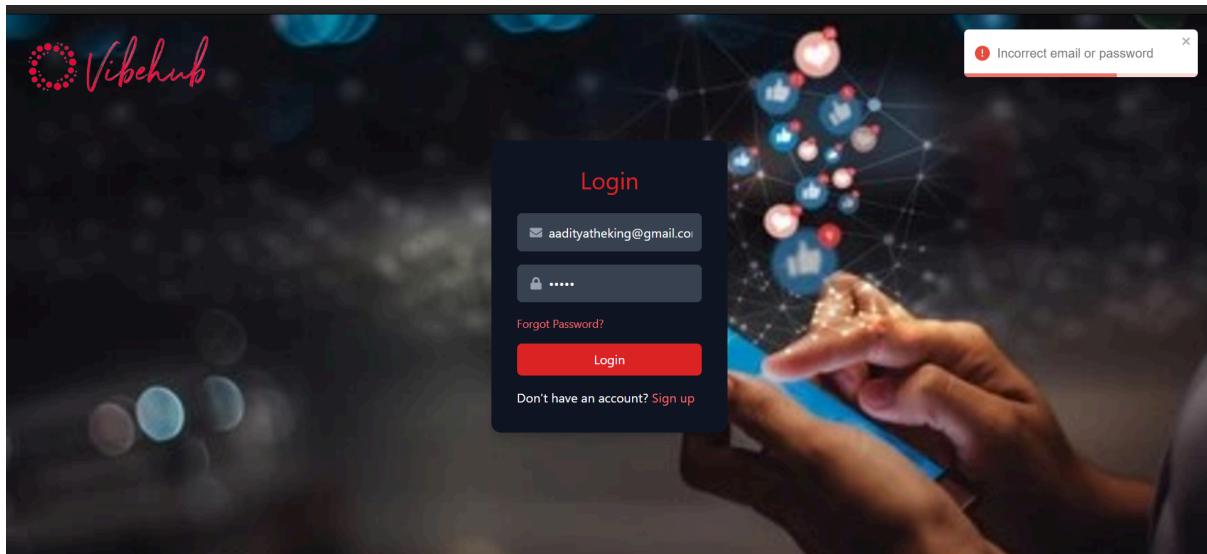
Output : Invalid



Email id:aadityatheking@gmail.com

Password : abcA12*

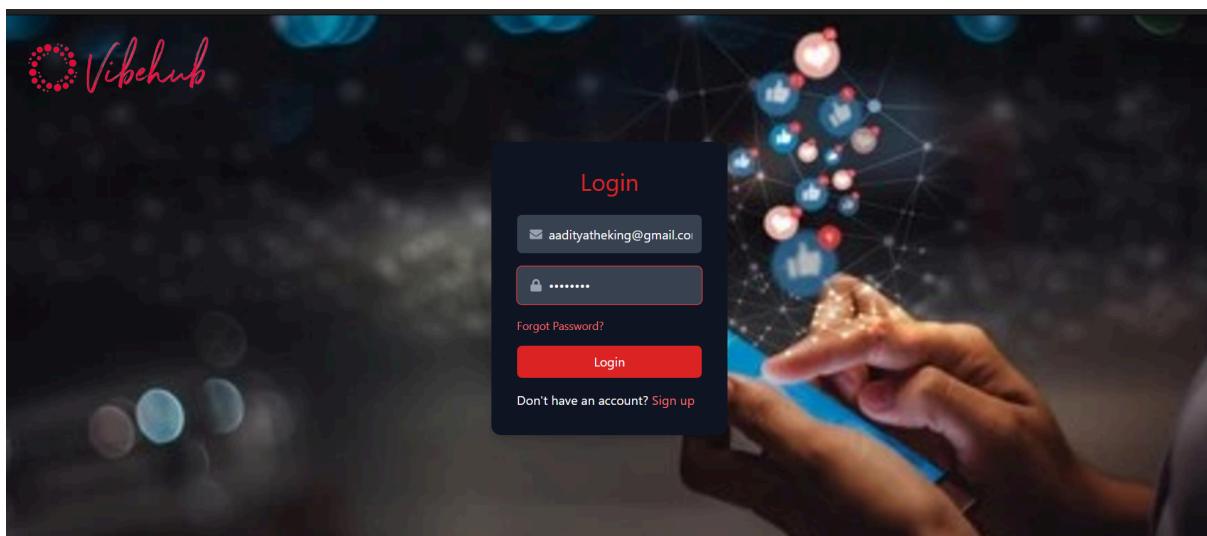
Output : Invalid



Email id: aadityatheking37@gmail.com

Password : Abc123**abc

Output : Valid

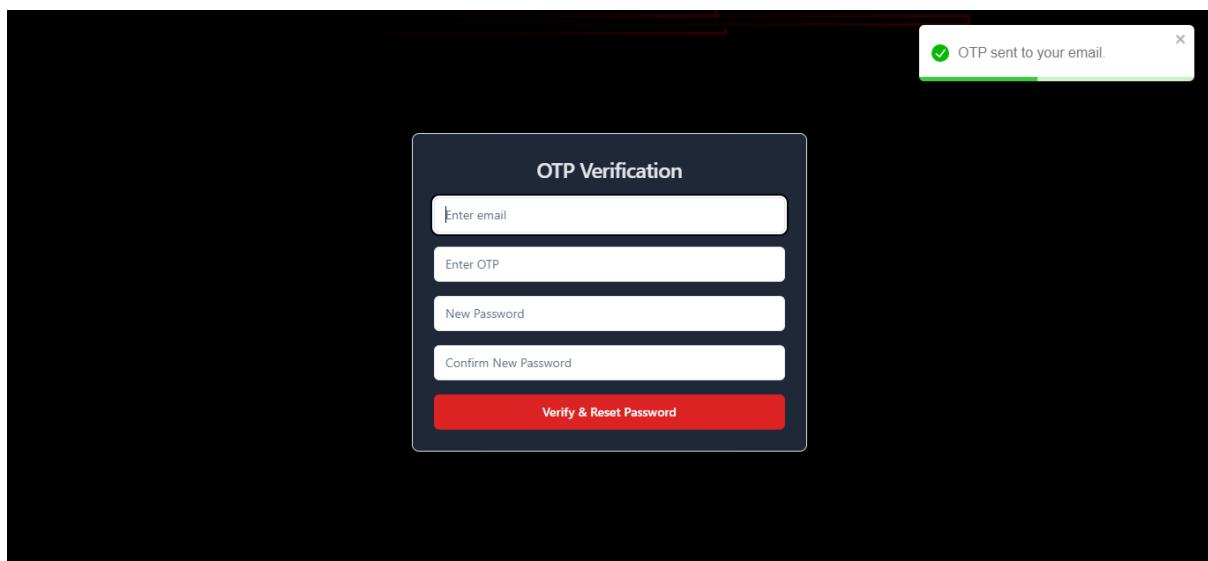


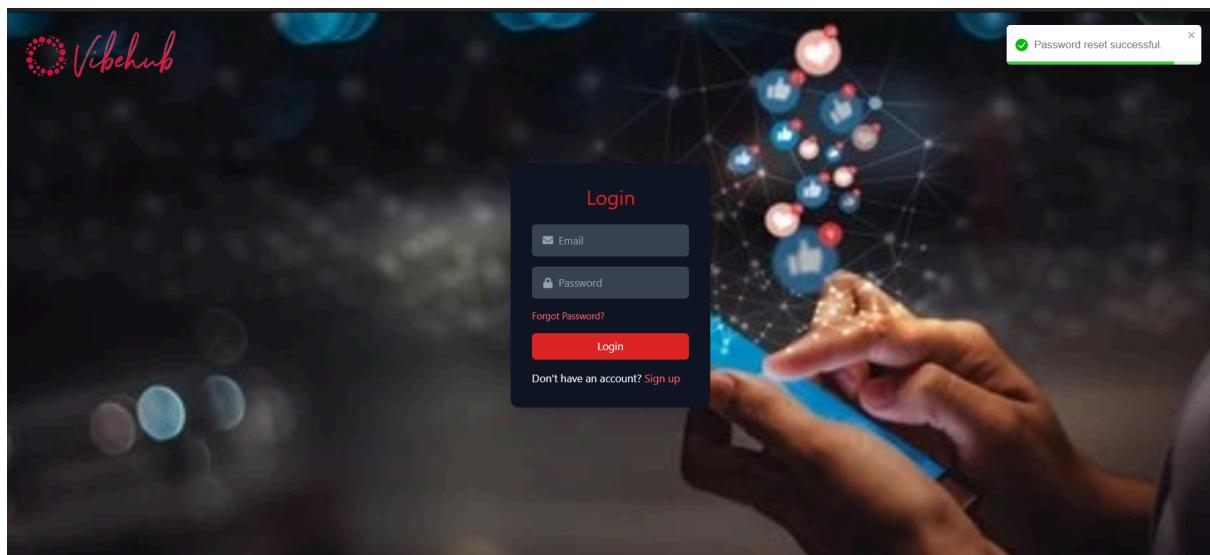


3. Forgot password

Email id : aadityatheking37@gmail.com

Output : valid





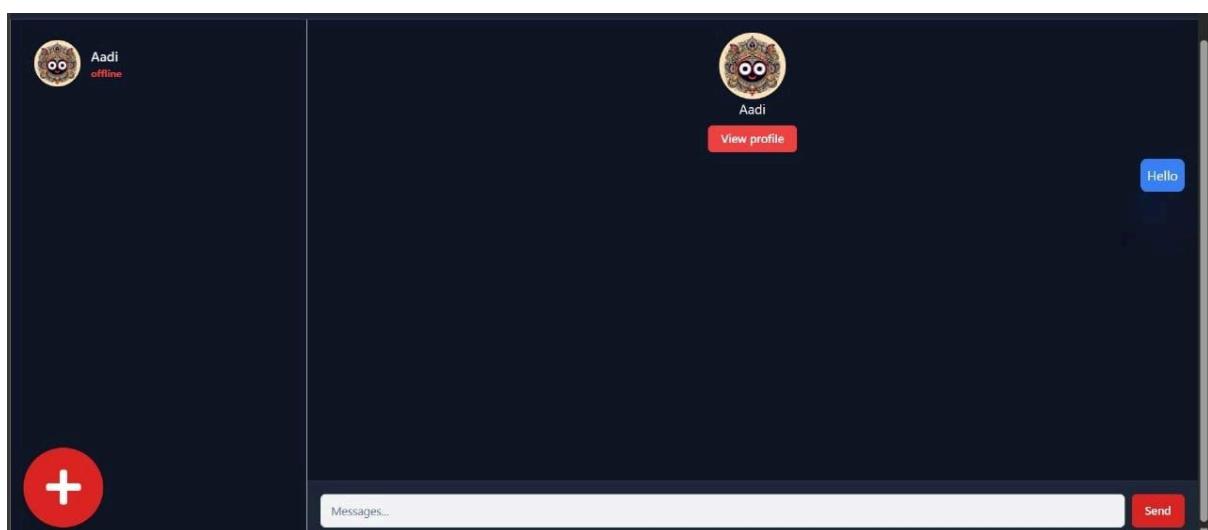
4. Messages

Message : hello

To:Aadi

From:vatsal123

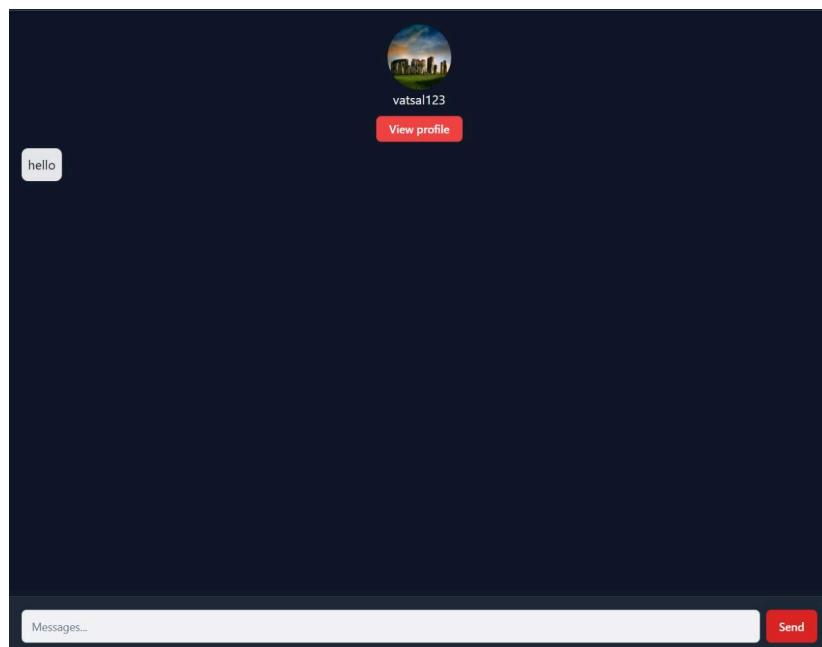
Aadi's Side



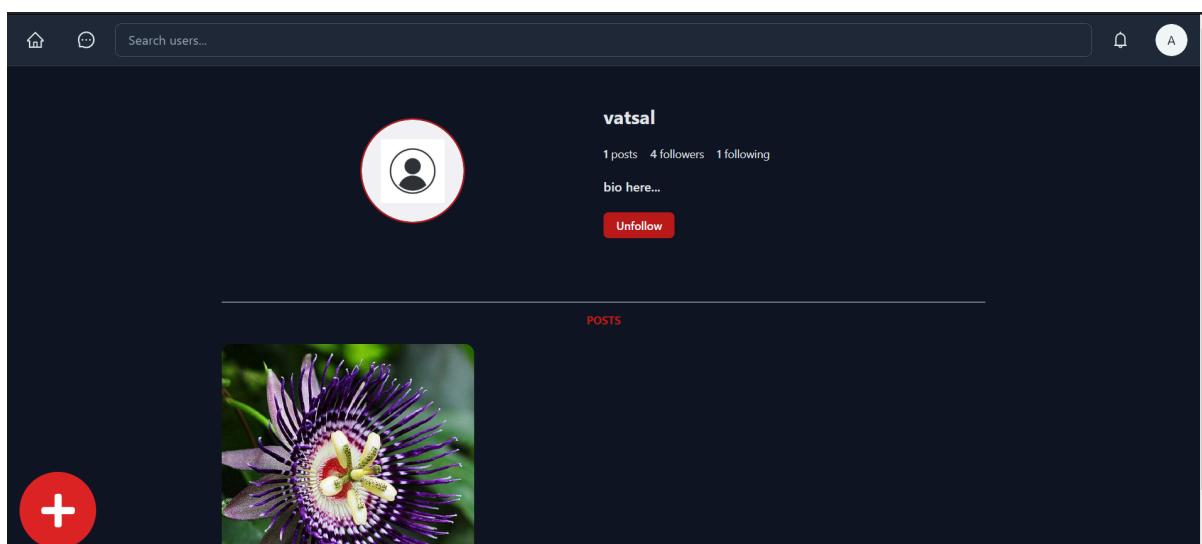
Message : hello

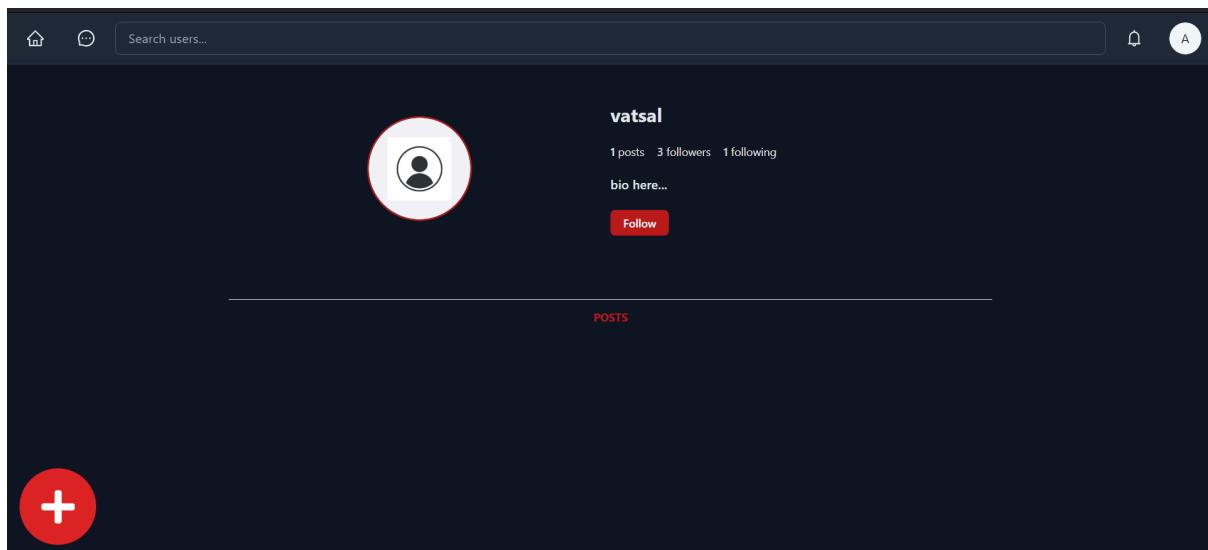
To:vatsal123

From:Aadi

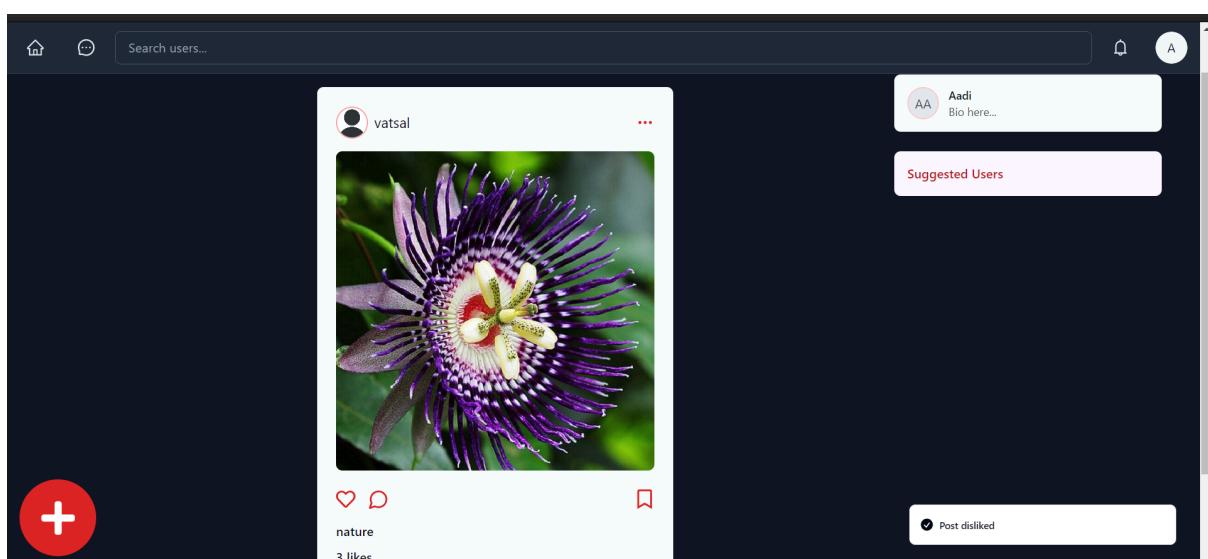
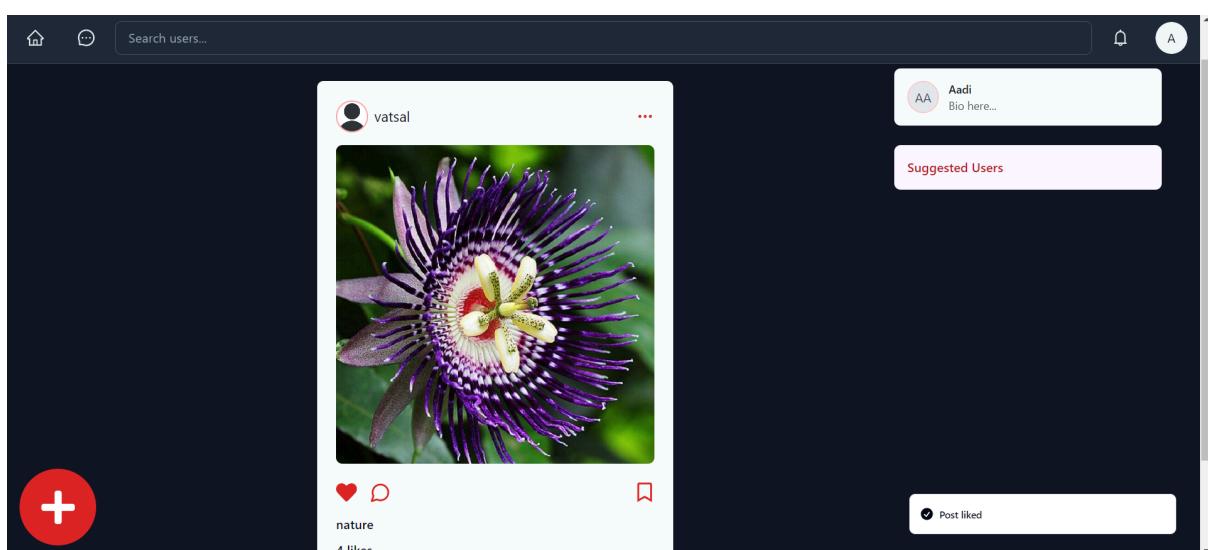


5. Follow/Unfollow user



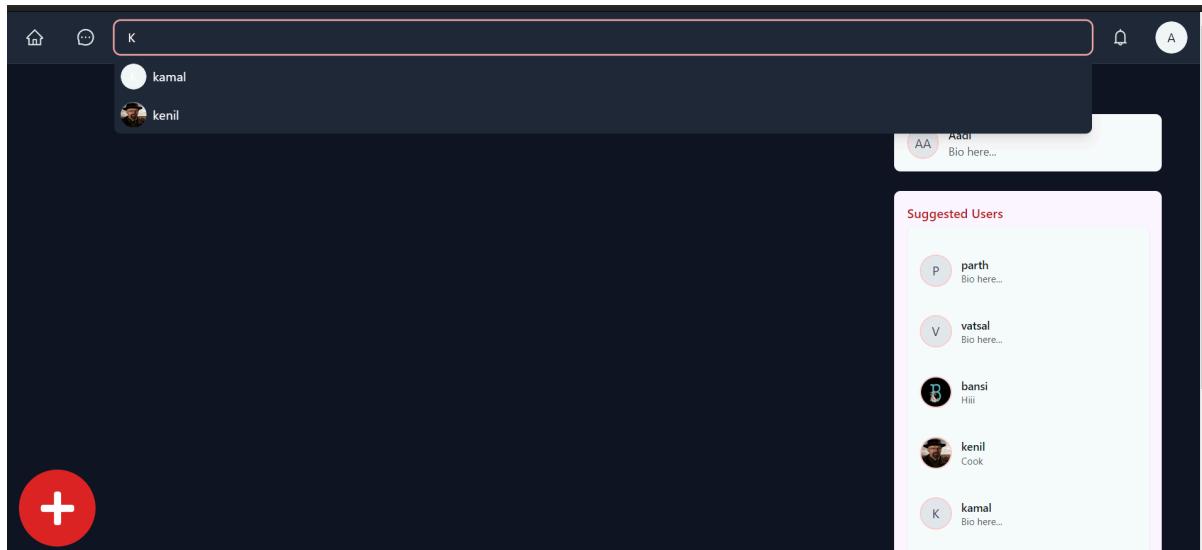


6. Like/Dislike post :



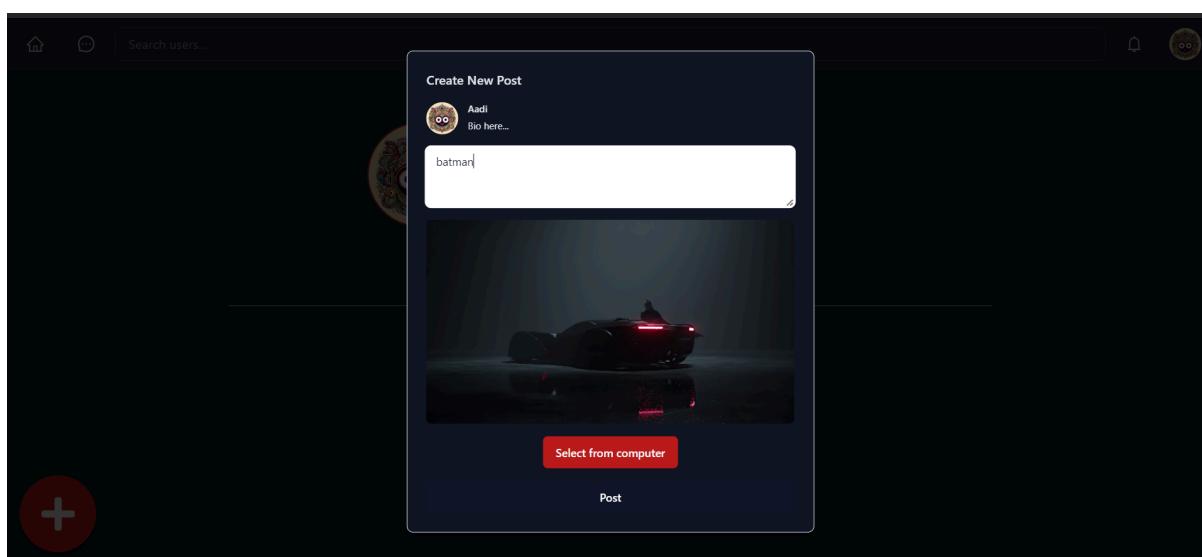
7. Search

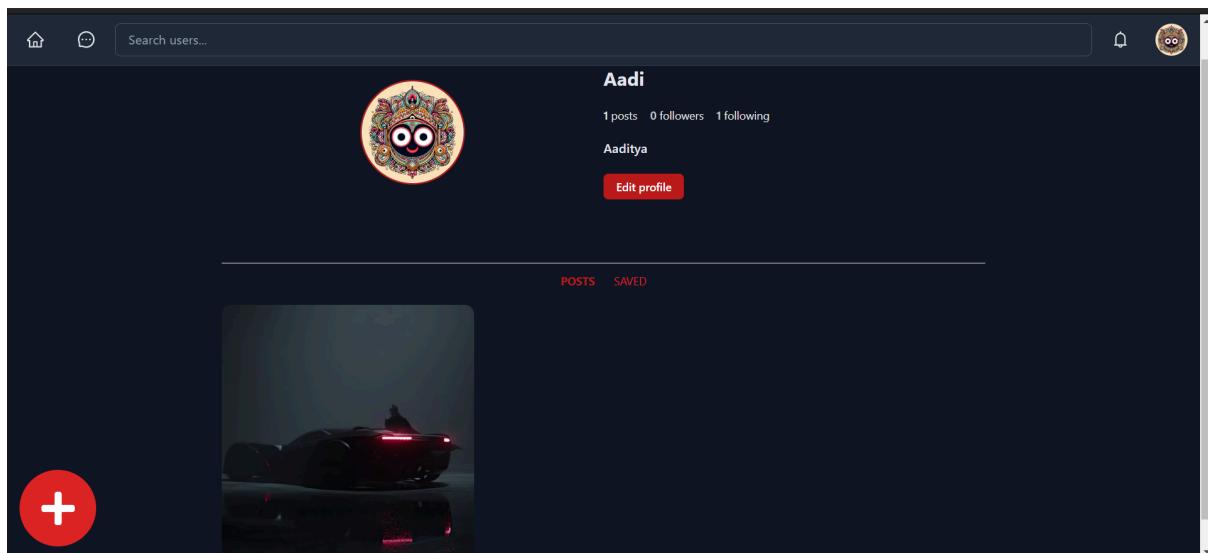
Username: "v"



8. Create Post

Caption : batman



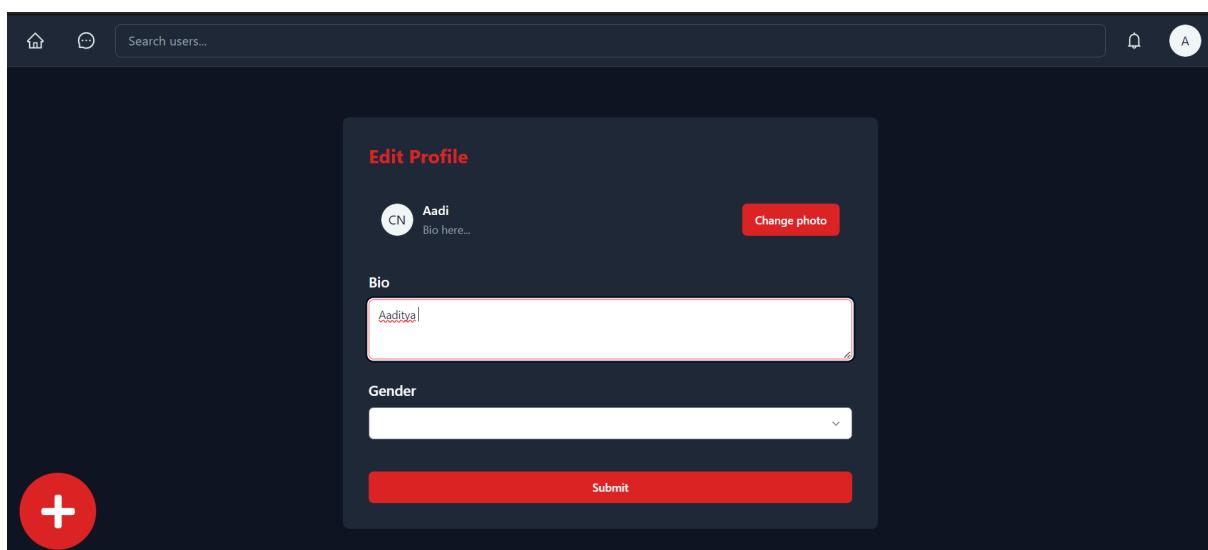


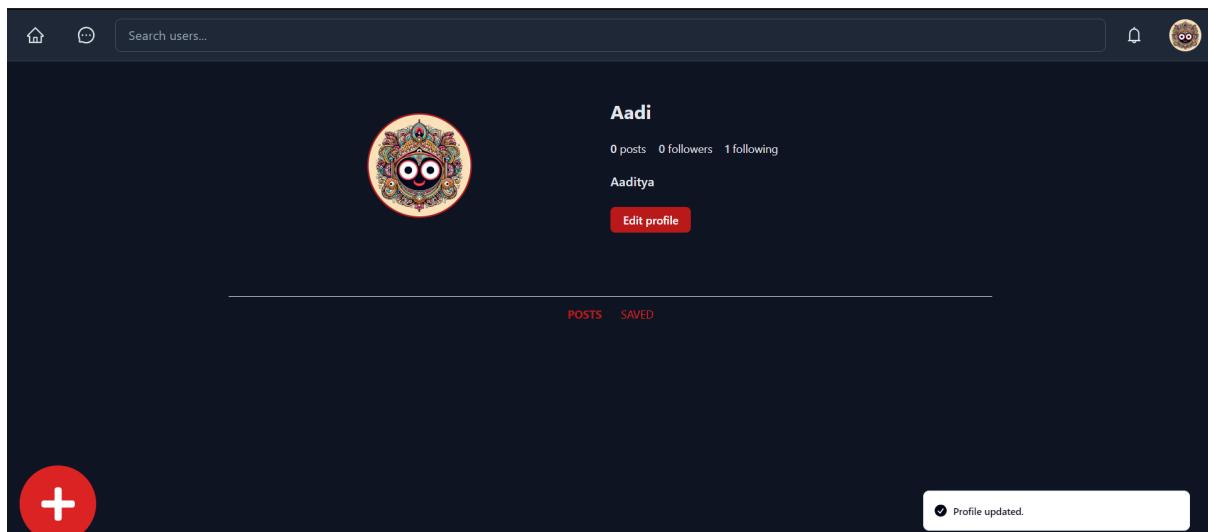
9.Edit Profile:

Profile Picture : “ “

Bio : Aaditya

Gender :Male





10.Notification :

