

User Acceptance Testing (UAT) Template

Date	17/7/2024
Team ID	SWTID1719938419
Project Name	Social Media App (MERN)
Maximum Marks	

Project Overview:

Project Name: Social Media App (MERN)

Project Description:

This project is a social media application clone designed to replicate the core functionalities of popular social media platforms. It aims to provide users with an interactive and engaging experience, allowing them to connect, share, and interact with each other.

Key Features

1. User Authentication: Secure user login and registration using JWT (JSON Web Tokens) and bcrypt for password hashing.
2. Profile Management: Users can view and edit their profiles, update personal information, and manage account settings.
3. Post Creation and Interaction: Users can create, edit, and delete posts, as well as like, comment, and share posts.
4. Real-Time Updates: Real-time notifications and updates using Socket.io to ensure users stay informed about interactions and new content.
5. Data Display: Dynamic data display using APIs to fetch and render user data, posts, and other relevant information.

Project Version: 1

Testing Period:

1 Week

Testing Scope :

List of Features and Functionalities to be Tested

1. User Authentication

- Registration
- Login
- Logout
- Password reset

2. User Profile Management

- View profile
- Edit profile details (name, email, bio, profile picture)
- Change password

3. Post Management

- Create a new post
- Edit an existing post
- Delete a post
- View a list of posts
- View post details

4. Interactions

- Like a post
- Comment on a post
- Delete a comment
- View comments on a post

5. Real-Time Features

- Real-time notifications (e.g., for likes, comments)
- Real-time updates of posts and comments using Socket.io

6. Dashboard

- Display user-specific data and statistics oShow recent activity
- Provide quick links to key features

7. Data Display

- Fetch and display data from backend APIs
- Handle loading states and errors

8. Navigation and Routing

- Correct rendering of components based on URL paths
- Navigation between different pages (home, dashboard, profile)
- Redirects and guards for authenticated/unauthenticated users

List of User Stories or Requirements to be Tested

- a. As a user, I want to register for an account so that I can access the application.
 - Successful registration
 - Error handling for duplicate emails, weak passwords, etc.
- b. As a user, I want to log in to my account so that I can access my personalized dashboard.
 - Successful login with valid credentials
 - Error handling for invalid credentials
- c. As a user, I want to log out of my account so that I can secure my session.
 - Successful logout
 - Redirection to the login page after logout
- d. As a user, I want to view and edit my profile so that I can update my personal information.
 - Display current profile information
 - Successful update of profile details
 - Error handling for invalid inputs
- e. As a user, I want to create, edit, and delete posts so that I can share content with others.
 - Successful creation of a new post
 - Successful editing of an existing post
 - Successful deletion of a post

- f. As a user, I want to interact with posts by liking and commenting so that I can engage with content.
 - Liking a post and updating the like count
 - Adding a comment to a post
 - Deleting a comment from a post
 - Viewing all comments on a post
 - g. As a user, I want to receive real-time notifications for interactions on my posts so that I am informed about engagement.
 - Receiving notifications for likes and comments
 - Real-time updates for new comments and likes
 - h. As a user, I want to view my dashboard to see an overview of my activity and statistics.
 - Displaying user-specific data and recent activity
 - Providing quick links to key features
- 2. As a user, I want to navigate through the application seamlessly so that I can easily access different sections.
 - Correct routing and rendering of components based on URL paths
 - Navigation between home, dashboard, and profile pages
 - Redirects and guards for authenticated/unauthenticated users

Testing Environment:

1. URL/Location: http://localhost:5000
2. Credentials:
 1. Username: testuser@example.com
 2. Password: TestPassword123

Test Cases:

Test Case ID	Test Scenario	Test Steps	Expected Result	Actual Result	Pass/Fail
TC-001	User Registration	<ol style="list-style-type: none">1. Navigate to the registration page.2. Fill in the registration form with valid details.3. Click on the "Register" button.	User should be able to register successfully.	Registration successful. User redirected to dashboard.	PASS
TC 002	User Login	<ol style="list-style-type: none">1. Navigate to the login page.2. Enter valid username password.	User should be able to log in successfully.	Login successful. User redirected to dashboard.	Pass
TC 003	User Profile View	<ol style="list-style-type: none">1. Navigate to the login page.2. Navigate to the profile page.	User should be able to view their profile details.	Profile details displayed correctly.	Pass

TC-004	User Profile edit	1.Login into the Application 2.Navigate to profile View 3.edit profile information	User should be able to update their profile details.	Profile updated success fully. Changes reflected on the profile page.	PASS
TC -005	POST CREATION	1. Log in to the application. 2. Navigate to post Creation page 3. fill the details	User should be able to create a new post.	feed and displayed.	PASS

Bug Tracking:

Bug ID	Bug Description	Steps to reproduce	Severity	Status	Additional feedback
BG 001	Registration form does not validate email format	1. Navigate to the registration page. 2. Enter an invalid email format. 3. Click on the "Register" button.	Medium	Open	Email validation should follow standard email format.
BG 002	Login button does not respond after clicking	1. Navigate to the login page. 2. Enter valid credentials 3. Click on the "Login" button.	High	In Progress	Check if the onClick event handler is properly set.

BG 003	Profile picture does not update after upload	1. Log in to the application. 2. Navigate to the profile page. 3. Upload a new profile picture and save changes.	Medium	OPEN	Ensure the image upload API is functioning correctly.
BG 004	Like count does not update in real-time	1. Log in to the application. 2. Navigate to the post page. 3. Click on the "Like" button for a post.	High	OPEN	Verify WebSocket connection for realtime updates.
BG 005	Dashboard statistics display incorrect data	1. Log in to the application. 2. Navigate to the dashboard page.	High	OPEN	Check the API responses for dashboard data.

Sign-off:

Tester Name: Sanjay

Date:17th July 2024

Signature: [Tester's Signature]

Notes:

3. Ensure that all test cases cover both positive and negative scenarios.
4. Encourage testers to provide detailed feedback, including any suggestions for improvement.
5. Bug tracking should include details such as severity, status, and steps to reproduce.
6. Obtain sign-off from both the project manager and product owner before proceeding with deployment.