

Generated Test Cases via mixtral-8x7b-32768 LLM Model and Zero-shot Prompting Technique

Data - 1

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

User Profiles

Feature description -

Each user can create a personalized profile that includes information such as their name, bio, profile picture, location, interests, and other relevant details. Profiles serve as a digital representation of individuals on the platform, allowing others to learn more about them.

Userstory title -

As a user, I want to create a personalized profile so that I can share my information with others on the platform.

Acceptance criteria -

['Users can add and edit their name, bio, profile picture, location, and interests.', 'The profile picture should support standard image formats.', 'Users can set privacy settings for each field.', 'The profile must be saved successfully.', 'Users should receive a confirmation upon saving their profile.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC1-UP_PersonalizedProfile

Test Case Title: Verify that users can add and edit their name, bio, profile picture, location, and interests.

Test Suite: User Profiles

Test Priority: High

Preconditions:

- User is logged in.
- User navigates to their profile settings page.

Test Data: Test user data (name, bio, profile picture, location, interests)

Test Steps:

1. Access the profile settings page.
2. Enter the test user data into the respective fields.
3. Save the changes.
4. Verify that the changes are reflected in the user's profile.
5. Edit the test user data.
6. Save the changes again.
7. Verify that the edited changes are reflected in the user's profile.

Postconditions:

- User's profile is updated correctly.
- User remains logged in.

Expected Result:

- The system should allow users to add and edit their name, bio, profile picture, location, and interests.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC2-UP_StandardImageFormats

Test Case Title: Verify that the profile picture supports standard image formats.

Test Suite: User Profiles

Test Priority: Medium

Preconditions:

- User is logged in.
- User navigates to their profile settings page.

Test Data: Test image files in various standard formats (JPG, PNG, GIF)

Test Steps:

1. Access the profile settings page.
2. Click on the profile picture upload button.
3. Select a test image file from the available formats.
4. Upload the image.
5. Verify that the uploaded image is displayed as the profile picture.

Postconditions:

- User's profile picture is updated with the selected image.
- User remains logged in.

Expected Result:

- The system should support standard image formats (JPG, PNG, GIF) for profile pictures.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Compatibility Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC3-UP_PrivacySettings

Test Case Title: Verify that users can set privacy settings for each field.

Test Suite: User Profiles

Test Priority: Medium

Preconditions:

- User is logged in.
- User navigates to their profile settings page.

Test Data: N/A

Test Steps:

1. Access the profile settings page.
2. Set privacy levels (Public, Friends, Only Me) for name, bio, profile picture, location, and interests.
3. Save the changes.
4. Verify that the privacy settings are correctly applied.

Postconditions:

- User's privacy settings are updated.
- User remains logged in.

Expected Result:

- The system should allow users to set privacy settings for each field.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Security Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC4-UP_ProfileSave

Test Case Title: Verify that the profile is saved successfully.

Test Suite: User Profiles

Test Priority: Medium

Preconditions:

- User is logged in.
- User navigates to their profile settings page.

Test Data: Test user data (name, bio, profile picture, location, interests)

Test Steps:

1. Access the profile settings page.
2. Enter the test user data into the respective fields.
3. Save the changes.
4. Verify that a confirmation message is displayed upon saving the profile.

Postconditions:

- User's profile is saved with the provided user data.
- User remains logged in.

Expected Result:

- The system should save the user's profile with the entered data and display a confirmation message.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC5-UP_NegativeName

Test Case Title: Verify that users cannot enter negative values for their name.

Test Suite: User Profiles

Test Priority: Low

Preconditions:

- User is logged in.
- User navigates to their profile settings page.

Test Data: Negative value (-1)

Test Steps:

1. Access the profile settings page.
2. Enter a negative value (-1) into the name field.
3. Save the changes.

Postconditions:

- User remains logged in.

Expected Result:

- The system should not accept negative values for the name field and display an error message.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing, Negative Testing, Data-Driven Testing

Test Case Approach: Negative

Data - 2

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

News Feed

Feature description -

The news feed is a central feature of the social media platform where users can view a curated stream of

content from accounts they follow or are connected to. It typically displays posts, updates, photos, videos, and other media shared by friends, family, brands, and other users in chronological or algorithmically determined order.

Userstory title -

As a user, I want to see a curated news feed so that I can stay updated with content from my connections.

Acceptance criteria -

['The news feed displays posts from followed accounts.', 'Content is shown in chronological or algorithmic order.', 'Users can interact with posts (like, comment, share).', 'The feed updates in real-time.', 'Users can filter content based on type or topic.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC-001-NEWSFEED-DISPLAY-POSTS

Test Case Title: Verify that the news feed displays posts from followed accounts.

Test Case Description: As a user, I want to ensure that my news feed displays content from the accounts I follow in the correct order.

Test Suite: News Feed Module

Test Priority: High

Preconditions:

- * User is logged into the social media platform.
- * User has followed one or more accounts.

Test Data: Accounts followed by the user.

Test Steps:

1. Log in to the social media platform.
2. Navigate to the news feed.
3. Verify that posts from the accounts followed by the user are visible.

Postconditions:

1. The news feed displays content from followed accounts.

Expected Result: The news feed shows posts from the accounts followed by the user in chronological or algorithmic order.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-002-NEWSFEED-CONTENT-ORDER

Test Case Title: Verify that content is shown in chronological or algorithmic order.

Test Case Description: As a user, I want to ensure that the content on my news feed is displayed in a meaningful order that allows me to consume information efficiently.

Test Suite: News Feed Module

Test Priority: Medium

Preconditions:

* User is logged into the social media platform.

* User has followed one or more accounts.

Test Data: Accounts followed by the user.

Test Steps:

1. Log in to the social media platform.
2. Navigate to the news feed.
3. Verify that posts are displayed in chronological or algorithmic order.

Postconditions:

1. The news feed shows content in the correct order.

Expected Result: The news feed displays content either in chronological or algorithmic order, depending on the platform's settings or the user's preference.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-003-NEWSFEED-INTERACTION

Test Case Title: Verify that users can interact with posts (like, comment, share).

Test Case Description: As a user, I want to ensure that I can engage with content on my news feed by liking, commenting on, or sharing posts.

Test Suite: News Feed Module

Test Priority: High

Preconditions:

- * User is logged into the social media platform.
- * User has followed one or more accounts.

Test Data: Accounts followed by the user and a post from one of the accounts.

Test Steps:

1. Log in to the social media platform.
2. Navigate to the news feed.
3. Select a post from one of the accounts followed by the user.
4. Like, comment on, or share the post.
5. Verify that the interaction is recorded and visible.

Postconditions:

1. The user's interaction with the post is visible to the user.

Expected Result: The user can like, comment on, or share posts, and these interactions are recorded and visible in the user interface.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-004-NEWSFEED-REAL-TIME-UPDATE

Test Case Title: Verify that the feed updates in real-time.

Test Case Description: As a user, I want to ensure that the news feed updates in real-time, allowing me to view new content as it becomes available.

Test Suite: News Feed Module

Test Priority: Medium

Preconditions:

- * User is logged into the social media platform.
- * User has followed one or more accounts.
- * One or more of the followed accounts have posted new content.

Test Data: Accounts followed by the user and new content from one of the accounts.

Test Steps:

1. Log in to the social media platform.
2. Navigate to the news feed.
3. Create new content on one of the accounts followed by the user.
4. Verify that the new content appears in the user's news feed without requiring a manual refresh.

Postconditions:

1. The new content is visible in the user's news feed.

Expected Result: The news feed updates in real-time, displaying new content as it becomes available.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC-005-NEWSFEED-CONTENT-FILTERING

Test Case Title: Verify that users can filter content based on type or topic.

Test Case Description: As a user, I want to ensure that I can filter the content on my news feed based on specific criteria, such as post type or topic.

Test Suite: News Feed Module

Test Priority: Medium

Preconditions:

- * User is logged into the social media platform.
- * User has followed one or more accounts.

Test Data: Accounts followed by the user and various post types.

Test Steps:

1. Log in to the social media platform.
2. Navigate to the news feed.
3. Use the filtering options to narrow down the displayed content.
4. Verify that the filtered content matches the selected criteria.

Postconditions:

1. The filtered content matches the user's selected criteria.

Expected Result: The news feed allows users to filter content based on type or topic, and the filtered content matches the selected criteria.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 3

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

Messaging

Feature description -

Messaging enables direct communication between users through private messages or group chats. Users can send text messages, multimedia files, emojis, stickers, and other forms of communication in real-time or asynchronously, fostering private conversations and connections.

Userstory title -

As a user, I want to send private messages so that I can communicate directly with my friends.

Acceptance criteria -

['Users can send text messages, images, and videos.', 'Messages can be sent in real-time.', 'Users can start private conversations or group chats.', 'Messages should show read receipts.', 'Users should be notified of new messages.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC-001-SM-Messaging-001

Test Case Title: Verify that users can send text messages in private conversations

Test Case Description: This test case aims to validate the user's ability to send text messages in private conversations.

Test Suite: Messaging

Test Priority: High

Preconditions:

- User must be logged in
- A friend must be present in the friend list

Test Data: Friend's username

Test Steps:

1. Navigate to the messaging section
2. Choose a friend from the friend list
3. Enter a text message in the input field
4. Click the "Send" button

Postconditions:

- The text message should be visible in the chat window

Expected Result: Text message is successfully sent and displayed in the chat window

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-002-SM-Messaging-002

Test Case Title: Verify that users can send multimedia files in private conversations

Test Case Description: This test case verifies the user's ability to send images and videos in private conversations.

Test Suite: Messaging

Test Priority: High

Preconditions:

- User must be logged in
- A friend must be present in the friend list

Test Data: Image or video file

Test Steps:

1. Navigate to the messaging section
2. Choose a friend from the friend list
3. Click the "Attach Media" button
4. Select an image or video file
5. Click the "Send" button

Postconditions:

- The selected multimedia file should be displayed in the chat window

Expected Result: Selected multimedia file is successfully sent and displayed in the chat window

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-003-SM-Messaging-003

Test Case Title: Verify that users receive notifications for new messages

Test Case Description: This test case tests the functionality of receiving notifications when new messages are received.

Test Suite: Messaging

Test Priority: Medium

Preconditions:

- User must be logged in
- A friend must be present in the friend list
- Notifications must be enabled

Test Data: No test data needed

Test Steps:

1. Navigate to the messaging section
2. Choose a friend from the friend list
3. Send them a text message or multimedia file
4. Wait for the friend to respond

Postconditions:

- A notification should appear, indicating a new message

Expected Result: User receives a notification when a new message is received

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-004-SM-Messaging-004

Test Case Title: Verify that messages show read receipts

Test Case Description: This test case ensures that messages show read receipts when the recipient has viewed the message.

Test Suite: Messaging

Test Priority: Medium

Preconditions:

- User must be logged in
- A friend must be present in the friend list

Test Data: No test data needed

Test Steps:

1. Navigate to the messaging section
2. Choose a friend from the friend list
3. Send them a text message or multimedia file
4. Wait for the friend to view the message
5. Observe the message status

Postconditions:

- The message status should update to "Read" once the friend has viewed it

Expected Result: Messages show read receipts when the recipient has viewed the message

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC-005-SM-Messaging-005

Test Case Title: Verify that users can start group chats

Test Case Description: This test case confirms the ability for users to initiate group chats with multiple recipients.

Test Suite: Messaging

Test Priority: Medium

Preconditions:

- User must be logged in
- At least 3 friends must be present in the friend list

Test Data: Friend usernames

Test Steps:

1. Navigate to the messaging section
2. Click the "New Group Chat" button
3. Select at least three friends from the friend list
4. Enter a group chat name in the input field
5. Click the "Create" button

Postconditions:

- The new group chat should be visible in the user's chat list

Expected Result: User can create a group chat and it is displayed in the chat list

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 4

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles,

privacy settings, notifications, and analytics.

Feature name -

Content Sharing

Feature description -

Users can share various types of content with their followers or the public, including text posts, photos, videos, links, articles, and documents. Content sharing fosters engagement, creativity, and self-expression among users, allowing them to share their thoughts, experiences, and interests with others.

Userstory title -

As a user, I want to share posts so that I can express my thoughts and interests with others.

Acceptance criteria -

['Users can create and share text posts, photos, and videos.', 'Content can be shared with followers or publicly.', 'Users can edit or delete their posts.', 'Posts should support hashtags and tagging.', 'Users should receive feedback through likes, comments, and shares.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that users can create and share text posts

Test Case Description: This test case verifies that registered users are able to create a new text post and share it with their followers or publicly.

Test Suite: Content Sharing Suite

Test Priority: High

Preconditions:

- User is logged in
- User is on the feed or profile page

Test Data: "This is a sample text post."

Test Steps:

1. Navigate to the feed or profile page

2. Click on the "Create Post" button
3. Select the "Text" option
4. Input the test data into the text box
5. Choose the desired audience (followers or public)
6. Click on the "Post" button

Postconditions:

- The new text post appears on the user's feed and/or profile
- The post is visible to the selected audience

Expected Result: The test case passes if the user can create and share a new text post successfully.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that users can share photos

Test Case Description: This test case checks if users can upload and share photos with their followers or publicly.

Test Suite: Content Sharing Suite

Test Priority: High

Preconditions:

- User is logged in
- User is on the feed or profile page

Test Data: Sample photo file

Test Steps:

1. Navigate to the feed or profile page
2. Click on the "Create Post" button

3. Select the "Photo" option
4. Choose the test data (sample photo file)
5. Input a caption for the photo (optional)
6. Choose the desired audience (followers or public)
7. Click on the "Post" button

Postconditions:

- The new photo post appears on the user's feed and/or profile
- The post is visible to the selected audience

Expected Result: The test case passes if the user can upload and share a new photo post successfully.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users can edit or delete their posts

Test Case Description: This test case verifies that users can modify or remove their own posts after they have been shared.

Test Suite: Content Sharing Suite

Test Priority: Medium

Preconditions:

- User is logged in
- User has at least one post

Test Data: N/A

Test Steps:

1. Navigate to the user's feed or profile
2. Locate a post created by the user

3. Click on the three-dot menu for the post
4. Select "Edit" or "Delete"
5. If editing, modify the post and click "Save"
6. If deleting, confirm the action

Postconditions:

- The post is updated if the "Edit" option was selected
- The post is removed if the "Delete" option was selected

Expected Result: The test case passes if the user can successfully edit or delete their own posts.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that posts support hashtags and tagging

Test Case Description: This test case ensures that users can include hashtags and tag other users in their posts.

Test Suite: Content Sharing Suite

Test Priority: Medium

Preconditions:

- User is logged in
- User is on the feed or profile page

Test Data: #samplehashtag @sampleuser

Test Steps:

1. Navigate to the feed or profile page
2. Click on the "Create Post" button
3. Create a text post with the test data (hashtag and tag)
4. Choose the desired audience (followers or public)

5. Click on the "Post" button

Postconditions:

- The new post appears on the user's feed and/or profile
- The post includes the hashtag and tag

Expected Result: The test case passes if the user can create a post with hashtags and tags other users successfully.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that users receive feedback through likes, comments, and shares

Test Case Description: This test case checks if users can interact with shared content by liking, commenting, and sharing posts.

Test Suite: Content Interaction Suite

Test Priority: Low

Preconditions:

- User is logged in
- Another user has shared a post

Test Data: N/A

Test Steps:

1. Navigate to the feed page
2. Locate a post shared by another user
3. Interact with the post by liking, commenting, and/or sharing

Postconditions:

- The interaction is recorded and visible on the post

Expected Result: The test case passes if the user can like, comment, and share posts successfully.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 5

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

Privacy Settings

Feature description -

Privacy settings allow users to control who can view their profile, posts, and other activity on the platform. Users can choose between public, private, or restricted privacy settings, determining who can see their content, send them friend requests, or interact with them.

Userstory title -

As a user, I want to set privacy settings so that I can control who sees my information.

Acceptance criteria -

['Users can set their profile to public, private, or restricted.', 'Privacy settings can be changed at any time.', 'Users can control visibility of individual posts.', 'The system should respect the privacy settings immediately.',

'Users should be notified of any changes in privacy settings.']

Generated Test Cases:

Test Case Creation Template for Privacy Settings Feature:

Test Case ID: TC-1

Test Case Title: Verify that users can set their profile to public, private, or restricted.

Test Case Description: This test case validates if users can change the privacy settings of their profile to public, private, or restricted.

Test Suite: User Profile Privacy Settings

Test Priority: High

Preconditions:

- User is logged in to the social media platform
- User navigates to the privacy settings page

Test Data: No test data needed

Test Steps:

1. Log in as a user.
2. Navigate to the privacy settings page.
3. Select the desired privacy setting (public, private, or restricted) for the profile.
4. Save the changes.

Postconditions: The user's profile privacy settings have been updated.

Expected Result: The user's profile should now be set to the selected privacy setting.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-2

Test Case Title: Verify that privacy settings can be changed at any time.

Test Case Description: This test case validates if users can change their privacy settings at any time.

Test Suite: User Profile Privacy Settings

Test Priority: Medium

Preconditions:

- User is logged in to the social media platform
- User has previously set profile privacy settings

Test Data: No test data needed

Test Steps:

1. Log in as a user.
2. Navigate to the privacy settings page.
3. Change the existing profile privacy setting.
4. Save the changes.

Postconditions: The user's profile privacy settings have been updated.

Expected Result: The user's profile privacy settings should have been updated to the new selection.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-3

Test Case Title: Verify that users can control visibility of individual posts.

Test Case Description: This test case validates if users can change the visibility of individual posts according to their preferences.

Test Suite: Post Privacy Settings

Test Priority: Medium

Preconditions:

- User is logged in to the social media platform
- User has previously created a post

Test Data: A test post with varying visibility settings

Test Steps:

1. Log in as a user.
2. Create a new post or select a previous post.
3. Change the visibility setting of the post (public, friends only, or custom).
4. Save the changes.

Postconditions: The selected post's visibility has been updated.

Expected Result: The post should now be visible according to the specified privacy setting.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-4

Test Case Title: Verify that the system should respect the privacy settings immediately.

Test Case Description: This test case ensures that the system recognizes and applies new privacy settings immediately.

Test Suite: User Profile Privacy Settings

Test Priority: High

Preconditions:

- User is logged in to the social media platform
- User navigates to the privacy settings page

Test Data: Test post with varying visibility settings

Test Steps:

1. Log in as a user.
2. Navigate to the privacy settings page.
3. Change the profile privacy settings.
4. Save the changes.
5. Create a new post and verify the visibility based on the updated privacy settings.

Postconditions: The user's profile privacy settings have been updated.

Expected Result: The visibility of the new post should match the updated privacy settings.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-5

Test Case Title: Verify that users are notified of any changes in privacy settings.

Test Case Description: This test case checks if users receive a notification when there are changes in their privacy settings.

Test Suite: User Notifications

Test Priority: Medium

Preconditions:

- User is logged in to the social media platform
- User navigates to the privacy settings page

Test Data: Test email address for notifications

Test Steps:

1. Log in as a user.
2. Navigate to the privacy settings page.
3. Change the profile privacy settings.
4. Save the changes.

Postconditions: The user's profile privacy settings have been updated.

Expected Result: A notification should be sent to the user's registered email address regarding the change in privacy settings.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 6

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

Notifications

Feature description -

Notifications alert users about new activity on the platform, such as likes, comments, mentions, friend requests, and event invitations. Notifications help users stay updated and engaged, prompting them to interact with the platform and respond to interactions from others.

Userstory title -

As a user, I want to receive notifications so that I am aware of new activity related to my profile.

Acceptance criteria -

['Users receive notifications for likes, comments, and shares.', 'Users are notified of new friend requests and mentions.', 'Notifications are displayed in real-time.', 'Users can customize notification preferences.', 'Notifications include a link to the relevant activity.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that users receive notifications for likes, comments, and shares.

Test Case Description: This test case verifies that users are notified about new likes, comments, and shares on their posts and content.

Test Suite: Notifications

Test Priority: High

Preconditions:

- User has an active account
- User has posted content (e.g., photo, video, or text post)

Test Data: Posts with images, text, or videos

Test Steps:

1. Log in as a registered user.
2. Create a new post with text, images, or videos.
3. Navigate to the Notifications tab.
4. Ask another user to like, comment on, or share the post.
5. Observe if a notification appears in real-time.

Postconditions:

- Notification is displayed in the Notifications tab

Expected Result:

- Users receive notifications for likes, comments, and shares.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that users are notified of new friend requests and mentions.

Test Case Description: This test case verifies that users receive notifications about new friend requests and mentions in posts or comments.

Test Suite: Notifications

Test Priority: High

Preconditions:

- User has an active account
- User has not yet approved a friend request

Test Data: Friend requests and mentions in posts or comments

Test Steps:

1. Log in as a registered user.
2. Request friendship with another user.
3. Create a new post and mention the user in the post.
4. Navigate to the Notifications tab.
5. Observe if a notification appears in real-time.

Postconditions:

- Notification is displayed in the Notifications tab

Expected Result:

- Users receive notifications for new friend requests and mentions.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that notifications are displayed in real-time.

Test Case Description: This test case verifies that notifications appear instantly or nearly instantly when new activity occurs.

Test Suite: Notifications

Test Priority: High

Preconditions:

- User has an active account
- User has posted content (e.g., photo, video, or text post)

Test Data: Posts with images, text, or videos

Test Steps:

1. Log in as a registered user.
2. Create a new post with text, images, or videos.
3. Ask another user to like, comment on, or share the post.
4. Immediately navigate to the Notifications tab.
5. Observe if a notification appears.

Postconditions:

- Notification is displayed in the Notifications tab

Expected Result:

- Notifications are displayed in real-time.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that users can customize notification preferences.

Test Case Description: This test case verifies that users can manage and modify their notification settings.

Test Suite: Notifications

Test Priority: Medium

Preconditions:

- User has an active account

Test Data: Notification settings

Test Steps:

1. Log in as a registered user.
2. Access the account settings.
3. Locate the Notification Preferences section.
4. Select or deselect specific notification types (e.g., likes, comments, friend requests).
5. Save the updated settings.
6. Perform actions that should trigger notifications based on the new settings.
7. Re-check the Notifications tab.

Postconditions:

- Notifications reflect the updated user preferences

Expected Result:

- Users can customize their notification preferences.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that notifications include a link to the relevant activity.

Test Case Description: This test case verifies that notifications contain a direct link to the post, comment, or

user related to the activity.

Test Suite: Notifications

Test Priority: Medium

Preconditions:

- User has an active account
- User has posted content (e.g., photo, video, or text post)

Test Data: Posts with images, text, or videos

Test Steps:

1. Log in as a registered user.
2. Create a new post with text, images, or videos.
3. Ask another user to like, comment on, or share the post.
- 4.

Navigate to the Notifications tab.

5. Click on the notification.
6. Check if the link directs the user to the relevant post, comment, or user profile.

Postconditions:

- Notifications contain a link to the relevant activity

Expected Result:

- Notifications include a link to the relevant activity.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

Search Functionality

Feature description -

Search functionality enables users to discover content, profiles, and topics of interest on the platform. Users can search for specific keywords, hashtags, accounts, or topics, helping them find relevant content and connect with others who share similar interests.

Userstory title -

As a user, I want to use search functionality to find specific content or profiles on the platform.

Acceptance criteria -

['Users can search for content using keywords or hashtags.', 'Search results include profiles, posts, and topics.', 'Results are displayed in order of relevance.', 'Users can filter search results by content type.', 'The search functionality provides suggestions as the user types.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that users can search for content using keywords or hashtags.

Test Case Description: This test case verifies that the search functionality allows users to input keywords or hashtags and returns relevant search results.

Test Suite: Search Functionality

Test Priority: High

Preconditions:

- * User is logged in.
- * User is on the platform's main page.

Test Data: Keyword or hashtag related to a post, profile, or topic.

Test Steps:

1. Enter a keyword or hashtag in the search bar.
2. Press the 'Search' button.
3. Observe the search results.

Postconditions:

- * Search results are displayed.

Expected Result: Search results include relevant posts, profiles, and topics containing the specified keyword or hashtag.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that search results include profiles, posts, and topics.

Test Case Description: This test case verifies that the search functionality returns a variety of results, including profiles, posts, and topics.

Test Suite: Search Functionality

Test Priority: High

Preconditions:

- * User is logged in.
- * User is on the platform's main page.

Test Data: Keyword or hashtag related to a post, profile, or topic.

Test Steps:

1. Enter a keyword or hashtag in the search bar.
2. Press the 'Search' button.
3. Observe the search results.

Postconditions:

* Search results are displayed.

Expected Result: Search results contain profiles, posts, and topics related to the input keyword or hashtag.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that results are displayed in order of relevance.

Test Case Description: This test case ensures that the search functionality ranks and displays search results based on relevance.

Test Suite: Search Functionality

Test Priority: Medium

Preconditions:

* User is logged in.

* User is on the platform's main page.

Test Data: Keyword or hashtag with multiple related search results.

Test Steps:

1. Enter a keyword or hashtag in the search bar.
2. Press the 'Search' button.
3. Observe the order of search results.

Postconditions:

* Search results are displayed.

Expected Result: Search results are displayed in descending order of relevance.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that users can filter search results by content type.

Test Case Description: This test case verifies that users have the ability to filter search results based on content type, such as posts, people, or topics.

Test Suite: Search Functionality

Test Priority: Medium

Preconditions:

* User is logged in.

* User is on the platform's main page.

Test Data: Keyword or hashtag with multiple related search results.

Test Steps:

1. Enter a keyword or hashtag in the search bar.
2. Press the 'Search' button.
3. Click on the 'Filter' button.
4. Select a content type from the filter options.
5. Observe the filtered search results.

Postconditions:

* Search results are filtered by content type.

Expected Result: The search displays filtered search results based on the selected content type.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that the search functionality provides suggestions as the user types.

Test Case Description: This test case confirms that the search bar provides real-time suggestions based on user input.

Test Suite: Search Functionality

Test Priority: Low

Preconditions:

- * User is logged in.
- * User is on the platform's main page.

Test Data: Keyword or hashtag related to a post, profile, or topic.

Test Steps:

1. Begin typing a keyword or hashtag into the search bar.
2. Observe the suggestions that appear in real-time.

Postconditions:

- * Suggestions are displayed in the search bar.

Expected Result: The search functionality provides relevant suggestions based on user input.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

Hashtags and Tagging

Feature description -

Hashtags and tagging allow users to categorize their content and make it more discoverable to others. Users can add relevant hashtags to their posts or tag other users, brands, or locations, increasing the visibility and reach of their content.

Userstory title -

As a user, I want to use hashtags and tagging so that I can categorize my content and reach a wider audience.

Acceptance criteria -

['Users can add hashtags to their posts.', 'Users can tag other profiles, brands, or locations.', 'Hashtags and tags are clickable and lead to related content.', 'Users can search for content using hashtags.', 'The system should suggest popular hashtags as users type.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC_HASHTAG_1

Test Case Title: Verify that users can add hashtags to their posts.

Test Case Description: Test the ability for users to add hashtags to their posts, confirming that the system correctly saves and displays the hashtags.

Test Suite: Hashtags and Tagging

Test Priority: High

Preconditions:

- User is logged in
- User has the ability to create a new post

Test Data: A valid post containing a hashtag (e.g. "This is a test post #testing")

Test Steps:

1. Log in as a user.
2. Create a new post.
3. Add a hashtag (e.g. "#testing") to the post content.
4. Save the post.
5. View the post on the user's profile and news feed.

Postconditions:

- The post is saved with the hashtag.
- The hashtag is displayed in the post on the user's profile and news feed.

Expected Result: The post displays the hashtag with a clickable link leading to related content.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC_HASHTAG_2

Test Case Title: Verify that users can tag other profiles, brands, or locations.

Test Case Description: Test the ability for users to tag other profiles, brands, or locations in their posts, confirming that the system correctly saves and displays the tags.

Test Suite: Hashtags and Tagging

Test Priority: High

Preconditions:

- User is logged in
- User has the ability to create a new post
- Another user, brand, or location exists

Test Data: A valid post containing a tag (e.g. "This is a test post @johndoe")

Test Steps:

1. Log in as a user.
2. Create a new post.
3. Tag another user, brand, or location (e.g. "@johndoe") in the post content.
4. Save the post.
5. View the post on the user's profile and news feed.

Postconditions:

- The post is saved with the tag.
- The tag is displayed in the post on the user's profile and news feed.

Expected Result: The post displays the tag with a clickable link leading to the tagged profile, brand, or location.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC_HASHTAG_3

Test Case Title: Verify that hashtags and tags are clickable and lead to related content.

Test Case Description: Test the clickability of hashtags and tags within posts, confirming that they direct users to the appropriate content.

Test Suite: Hashtags and Tagging

Test Priority: High

Preconditions:

- User is logged in
- A post containing a hashtag or tag exists

Test Data: A post with a hashtag or tag

Test Steps:

1. Log in as a user.
2. Navigate to a post containing a hashtag or tag.
3. Click on the hashtag or tag.

Postconditions:

- User is directed to a new page displaying related content for the hashtag or tag.

Expected Result: Hashtags and tags are clickable and redirect users to related content.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC_HASHTAG_4

Test Case Title: Verify that users can search for content using hashtags.

Test Case Description: Test the ability for users to search content through hashtags, confirming that the system returns relevant results.

Test Suite: Hashtags and Tagging

Test Priority: Medium

Preconditions:

- User is logged in
- Hashtags are associated with posts

Test Data: A valid hashtag (e.g. "#testing")

Test Steps:

1. Log in as a user.
2. Type the hashtag into the search bar.
3. Submit the search.

Postconditions:

- System returns relevant content containing the hashtag.

Expected Result: The system returns relevant content containing the hashtag.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC_HASHTAG_5

Test Case Title: Verify that the system suggests popular hashtags as users type.

Test Case Description: Test the system's ability to suggest popular hashtags while users are typing, ensuring that the suggestions are relevant.

Test Suite: Hashtags and Tagging

Test Priority: Medium

Preconditions:

- User is logged in

Test Data: Typing various keywords into the search or post content area

Test Steps:

1. Log in as a user.
2. Type a keyword into the search bar or post content area.
3. Observe if the system suggests popular hashtags related to the keyword.

Postconditions:

- Relevant hashtag suggestions appear.

Expected Result: The system suggests popular hashtags related to the keyword.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 9

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

Emojis and Reactions

Feature description -

Emojis and reactions provide users with expressive tools to react to posts and messages. Users can use emojis to convey emotions such as joy, sadness, love, or excitement, enhancing communication and engagement on the platform.

Userstory title -

As a user, I want to use emojis and reactions so that I can express my feelings about posts and messages.

Acceptance criteria -

['Users can react to posts with a variety of emojis.', 'Emojis can be used in comments and messages.', 'Users can see a count of reactions on posts.', 'The platform should support a wide range of emojis.', 'Users can remove or change their reactions.']

Generated Test Cases:

Test Case ID: TC01

Test Case Title: Verify that users can react to posts with a variety of emojis.

Test Case Description: Test the functionality of reacting to posts with different emojis available on the platform.

Test Suite: Post Reactions Suite

Test Priority: High

Preconditions:

- User must be logged in.
- A post must exist on the user's news feed.

Test Data: A selection of different emojis to react with.

Test Steps:

1. Navigate to the news feed.
2. Find a post to react to.
3. Click on the reaction button.
4. Select an emoji from the list.
5. Confirm the reaction.

Postconditions:

- The reaction should be visible below the post.
- The post's reaction count should increase by one.

Expected Result: Different emojis can be used as reactions to posts successfully.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC02

Test Case Title: Verify that emojis can be used in comments and messages.

Test Case Description: Test the ability to use emojis while composing and sending comments or messages.

Test Suite: Comment and Message Reactions Suite

Test Priority: High

Preconditions:

- User must be logged in.
- A post or conversation must exist for commenting or messaging.

Test Data: A selection of different emojis to use in comments or messages.

Test Steps:

1. Navigate to a post or conversation.
2. Click on the comment or reply input field.
3. Enter text and select an emoji.
4. Send the comment/message.

Postconditions:

- The emoji should appear correctly in the comment/message.

Expected Result: Emojis can be used in comments and messages successfully.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC03

Test Case Title: Verify that users can see a count of reactions on posts.

Test Case Description: Test the display of the total reaction count for each post along with the individual reactions.

Test Suite: Reaction Count Suite

Test Priority: Medium

Preconditions:

- User must be logged in.
- Posts with reactions must exist on the user's news feed.

Test Data: No test data needed.

Test Steps:

1. Navigate to the news feed.
2. Find a post with reactions.
3. Verify the reaction count below the post.

Expected Result: The post displays the correct reaction count, and individual reactions can be viewed.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC04

Test Case Title: Verify that users can remove or change their reactions.

Test Case Description: Test the user's ability to modify or delete their reactions on posts, comments, or messages.

Test Suite: Reaction Modification Suite

Test Priority: Medium

Preconditions:

- User must be logged in.
- Posts, comments, or messages with the user's reactions must exist.

Test Data: A previously placed reaction on a post, comment, or message.

Test Steps:

1. Navigate to the post, comment, or message with the user's reaction.
2. Click on the user's reaction.
3. Select an option to remove or change the reaction.
4. Confirm the action.

Expected Result: The reaction is successfully removed or changed.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC05

Test Case Title: Verify that the platform supports a wide range of emojis.

Test Case Description: Test the platform's compatibility with various emojis from different operating systems and devices.

Test Suite: Cross-Platform Emoji Compatibility Suite

Test Priority: Low

Preconditions:

- User must be logged in.
- Access to different devices or operating systems is required.

Test Data: A set of emojis from various platforms.

Test Steps:

1. Log in using different devices or operating systems.
2. Attempt to use the set of emojis in posts, comments, or messages.

Expected Result: The platform should support and display a wide range of emojis from different platforms correctly.

Severity: Trivial

Priority: Low

Type of Testing: Compatibility Testing, Cross-Browser Testing

Test Case Approach: Positive

Data - 10

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

Live Streaming

Feature description -

Live streaming enables users to broadcast real-time video content to their followers or the public. Users can share live events, performances, Q&A sessions, tutorials, and behind-the-scenes footage, allowing for authentic and interactive experiences with their audience.

Userstory title -

As a user, I want to broadcast live video so that I can share real-time events with my followers.

Acceptance criteria -

['Users can start and stop live streams.', 'Live streams can be public or limited to followers.', 'Viewers can interact with the streamer through comments and reactions.', 'The stream should display viewer count and live comments.', 'Users can save the live stream for later viewing.']

Generated Test Cases:

Test Case ID: TC001_LiveStreaming_StartStop

Test Case Title: Verify that users can start and stop live streams

Test Case Description: As a user, I want to start and stop live video streams to control the content I share with my followers.

Test Suite: Live Streaming

Test Priority: High

Preconditions:

- User is logged in
- User has a following or public profile

Test Data: No test data needed

Test Steps:

1. Navigate to the live streaming page
2. Click on the "Go Live" button
3. Verify that the live stream starts
4. Click on the "End Live Stream" button
5. Verify that the live stream stops

Postconditions:

- Live stream is no longer visible in the user's profile or news feed

Expected Result: Users can successfully start and stop live streams, allowing them to share real-time events with their followers.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002_LiveStreaming_Privacy

Test Case Title: Verify that users can choose between public and followers-only live streams

Test Case Description: As a user, I want to have the option to limit my live stream audience to my followers or make it public.

Test Suite: Live Streaming

Test Priority: High

Preconditions:

- User is logged in
- User has a following or public profile

Test Data: N/A

Test Steps:

1. Navigate to the live streaming page
2. Select the desired privacy option (Public or Followers Only)
3. Start a live stream
4. Verify that the live stream is accessible according to the selected privacy option

Postconditions:

- Live stream is accessible only to the intended audience

Expected Result: Users can start live streams that are either public or limited to their followers, ensuring they control their audience and content access.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003_LiveStreaming_Interaction

Test Case Title: Verify that viewers can interact with the streamer through comments and reactions

Test Case Description: As a viewer, I want to engage with the live stream by posting comments and reactions,

enhancing the interactive experience.

Test Suite: Live Streaming

Test Priority: High

Preconditions:

- User is logged in
- A live stream is in progress

Test Data: Any relevant text for comments and any available reactions

Test Steps:

1. Navigate to a live stream
2. Post a comment
3. Verify that the comment appears in the live stream's comment section
4. Add a reaction to the live stream
5. Verify that the reaction count updates accordingly

Postconditions:

- Comment and reaction data is saved for future reference

Expected Result: Viewers can interact with the live stream by posting comments and adding reactions, improving the overall user experience.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004_LiveStreaming_ViewerCountComments

Test Case Title: Verify that the live stream displays the viewer count and real-time comments

Test Case Description: As a viewer, I want to see the number of people watching the live stream and the real-time comments, fostering a sense of community and engagement.

Test Suite: Live Streaming

Test Priority: High

Preconditions:

- User is logged in
- A live stream is in progress

Test Data: N/A

Test Steps:

1. Navigate to a live stream
2. Verify that the viewer count is displayed and updates as viewers join or leave the live stream
3. Verify that real-time comments appear in the comment section and are displayed in chronological order

Postconditions: N/A

Expected Result: The live stream displays both the viewer count and real-time comments, allowing viewers to stay informed and engaged throughout the broadcast.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005_LiveStreaming_SaveForLater

Test Case Title: Verify that users can save live streams for later viewing

Test Case Description: As a user, I want to have the option to save a live stream for future viewing, so I don't miss any important content.

Test Suite: Live Streaming

Test Priority: High

Preconditions:

- User is logged in
- A live stream is in progress or has ended

Test Data: N/A

Test Steps:

1. Navigate to a live stream
2. Click on the "Save for Later" button
3. Verify that the live stream has been saved to the user's saved content or library

Postconditions:

- The live stream is saved for future access

Expected Result: Users can save live streams for later viewing, providing them with the ability to revisit content at their convenience.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 11

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

Stories

Feature description -

Stories are temporary posts that disappear after a certain period, typically 24 hours. Users can share ephemeral

content such as photos, videos, and text overlays in their stories, providing a more casual and spontaneous way to share moments and updates with their followers.

Userstory title -

As a user, I want to share stories so that I can share temporary moments with my followers.

Acceptance criteria -

['Users can create stories with photos, videos, and text overlays.', 'Stories disappear after 24 hours.', 'Users can see who viewed their stories.', 'Stories can be shared publicly or with selected followers.', 'Users can add stickers and filters to their stories.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC-001

Test Case Title: Verify that users can create stories with photos

Test Case Description: This test case checks if a user can successfully create and publish a story containing a photo.

Test Suite: Stories Module

Test Priority: High

Preconditions:

- User is logged in
- User has a profile picture

Test Data: A photo file saved on the device

Test Steps:

1. Tap the "Create Story" button
2. Select the "Add Photo" option
3. Choose a photo from the device's photo library
4. Apply any desired text overlays, stickers, or filters

5. Tap the "Share Story" button

Postconditions:

- Story is visible in the user's profile
- Story is visible in the news feed for approved followers

Expected Result: The story should be successfully created, displaying the chosen photo along with any applied text overlays, stickers, or filters.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-002

Test Case Title: Verify that stories disappear after 24 hours

Test Case Description: This test case checks if stories are automatically removed from the platform after 24 hours.

Test Suite: Stories Module

Test Priority: Medium

Preconditions:

- User has created a story with photos or videos

Test Data: No test data needed

Test Steps:

1. Create a story with photos or videos
2. Wait for 24 hours
3. Check the user's profile and news feed

Postconditions:

- Story is no longer visible in the user's profile
- Story is no longer visible in the news feed for approved followers

Expected Result: The story should be removed from the platform after 24 hours.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-003

Test Case Title: Verify that users can view who saw their stories

Test Case Description: This test case checks if users can view a list of followers who have viewed their stories.

Test Suite: Stories Module

Test Priority: High

Preconditions:

- User has created a story with photos or videos
- User has at least one follower who has viewed their story

Test Data: No test data needed

Test Steps:

1. Create a story with photos or videos
2. Wait for at least one follower to view the story
3. Tap the "Seen By" option below the story

Postconditions:

- List of viewers is displayed

Expected Result: A list of followers who have viewed the story should be displayed.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-004

Test Case Title: Verify that users can share stories publicly or with selected followers

Test Case Description: This test case checks if users can control the visibility of their stories by sharing them publicly or with chosen followers.

Test Suite: Stories Module

Test Priority: High

Preconditions:

- User is logged in
- User has at least one follower

Test Data: No test data needed

Test Steps:

1. Tap the "Create Story" button
2. Select the "Add Photo" or "Add Video" option
3. Choose a photo or video from the device's photo library
4. In the "Who Can See This?" section, select either "Public" or "Followers"
5. Tap the "Share Story" button

Postconditions:

- Story is visible to the selected audience

Expected Result: The story should be visible to the specified audience, either the public or selected followers.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC-005

Test Case Title: Verify that users can add stickers and filters to their stories

Test Case Description: This test case checks if users can add stickers and filters to their stories as desired.

Test Suite: Stories Module

Test Priority: Medium

Preconditions:

- User is logged in
- User has a photo or video for the story

Test Data: A photo or video file saved on the device

Test Steps:

1. Tap the "Create Story" button
2. Select the "Add Photo" or "Add Video" option
3. Choose a photo or video from the device's photo library
4. Tap the "Stickers" or "Filters" option
5. Select a sticker or filter and apply it to the story
6. Tap the "Share Story" button

Postconditions:

- Story is visible in the user's profile with the applied sticker or filter

Expected Result: The story should be visible in the user's profile with the selected sticker or filter.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 12

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

Analytics and Insights

Feature description -

Analytics and insights provide users with data and metrics about their activity, audience engagement, and content performance on the platform. Users can access information such as follower demographics, post reach, engagement metrics, and audience growth trends, helping them optimize their presence and strategy on the platform.

Userstory title -

As a user, I want to access analytics and insights so that I can understand my audience and optimize my content.

Acceptance criteria -

['Users can view data on follower demographics.', 'Post reach and engagement metrics are available.', 'Users can track audience growth trends.', 'Analytics are displayed in a user-friendly format.', 'Data can be filtered by time range and content type.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that users can view data on follower demographics

Test Case Description: As a user, I want to access analytics and insights about my follower demographics to understand my audience better.

Test Suite: Analytics and Insights

Test Priority: High

Preconditions:

- User is logged in
- User has followers

Test Data: No test data needed

Test Steps:

1. Log in as a user
2. Navigate to the Analytics and Insights section
3. Select the Follower Demographics tab
4. View the data presented (age, gender, location)

Postconditions: User has successfully viewed their follower demographics data

Expected Result: The follower demographics data is displayed accurately and in a user-friendly format

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that post reach and engagement metrics are available

Test Case Description: As a user, I want to access analytics and insights about my post reach and engagement

metrics to optimize my content strategy.

Test Suite: Analytics and Insights

Test Priority: High

Preconditions:

- User is logged in
- User has posted content

Test Data: No test data needed

Test Steps:

1. Log in as a user
2. Navigate to the Analytics and Insights section
3. Select the Post Reach and Engagement tab
4. View the data presented (impressions, likes, comments, shares)

Postconditions: User has successfully viewed their post reach and engagement metrics

Expected Result: The post reach and engagement metrics are displayed accurately and in a user-friendly format

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users can track audience growth trends

Test Case Description: As a user, I want to access analytics and insights about my audience growth trends to monitor my progress on the platform.

Test Suite: Analytics and Insights

Test Priority: High

Preconditions:

- User is logged in
- User has followers

Test Data: No test data needed

Test Steps:

1. Log in as a user
2. Navigate to the Analytics and Insights section
3. Select the Audience Growth Trends tab
4. View the data presented (follower count over time)

Postconditions: User has successfully viewed their audience growth trends

Expected Result: The audience growth trends data is displayed accurately and in a user-friendly format

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that analytics are displayed in a user-friendly format

Test Case Description: As a user, I want the analytics and insights to be displayed in a clear, easy-to-understand format so that I can quickly make informed decisions about my content strategy.

Test Suite: Analytics and Insights

Test Priority: High

Preconditions:

- User is logged in

Test Data: No test data needed

Test Steps:

1. Log in as a user
2. Navigate to the Analytics and Insights section

3. Review the data presentation format for each of the tabs (Follower Demographics, Post Reach and Engagement, Audience Growth Trends)

Postconditions: User has reviewed the data presentation format

Expected Result: The analytics and insights are displayed in a user-friendly, clear, and easy-to-understand format

Severity: Major

Priority: High

Type of Testing: Usability Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that data can be filtered by time range and content type

Test Case Description: As a user, I want to be able to filter my analytics and insights data by time range and content type so that I can analyze specific time periods and content types.

Test Suite: Analytics and Insights

Test Priority: High

Preconditions:

- User is logged in
- User has posted content

Test Data: No test data needed

Test Steps:

1. Log in as a user
2. Navigate to the Analytics and Insights section
3. Select a tab (Follower Demographics, Post Reach and Engagement, Audience Growth Trends)
4. Filter the data by time range and content type
5. Review the filtered data

Postconditions: User has filtered the data by time range and content type and reviewed the results

Expected Result: The data is filtered accurately according to the selected time range and content type

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 13

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

Events and Groups

Feature description -

Events and groups enable users to organize and participate in events, communities, or interest-based groups on the platform. Users can create events, RSVP to invitations, join groups, and participate in discussions, fostering community engagement and connection around shared interests.

Userstory title -

As a user, I want to join events and groups so that I can connect with people who share my interests.

Acceptance criteria -

['Users can create and join events.', 'Users can join interest-based groups.', 'Group discussions can be organized by topic.', 'Event details include date, time, location, and description.', 'Users can RSVP to events

and see the list of attendees.']

Generated Test Cases:

Test Case ID: TC-EG-001

Test Case Title: Verify that users can join interest-based groups

Test Case Description: Validate the ability for a user to browse, search for, and join an interest-based group on the Social Media Platform.

Test Suite: Events and Groups

Test Priority: High

Preconditions:

- * User is logged in.
- * User has access to the Social Media Platform.

Test Data:

- * User with valid login credentials.
- * Name or keyword related to the group.

Test Steps:

1. Navigate to the Groups section on the Social Media Platform.
2. Search for a group using the search bar or browse groups by category.
3. Select the desired group from the list of results.
4. Click the "Join Group" button on the group's page.
5. Confirm the action by clicking "Yes" on the confirmation dialog.

Postconditions:

- * User has joined the group.
- * User is displayed as a member on the group's page.

Expected Result:

- * User has successfully joined the group and can participate in group discussions.

Severity: Major

Priority: High

Type of Testing: Functional Testing, User Interface Testing

Test Case Approach: Positive

Test Case ID: TC-EG-002

Test Case Title: Verify that users can RSVP to events

Test Case Description: Validate the ability for a user to RSVP to an event, view their RSVP status, and see the list of attendees.

Test Suite: Events and Groups

Test Priority: High

Preconditions:

- * User is logged in.
- * User has access to the Social Media Platform.
- * Event exists on the platform.

Test Data:

- * User with valid login credentials.

Test Steps:

1. Navigate to the Events section on the Social Media Platform.
2. Search for an event by date, category, or location.
3. Select the desired event from the list of results.
4. Click the "RSVP" button on the event's page.
5. Confirm the action by clicking "Yes" on the confirmation dialog.
6. View the user's RSVP status on the event's page.
7. Check the list of attendees for the event.

Postconditions:

- * User's RSVP status is updated.
- * User can see the list of attendees for the event.

Expected Result:

* User has successfully RSVPed to the event and can view their RSVP status and the list of attendees.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-EG-003

Test Case Title: Verify that group discussions are organized by topic

Test Case Description: Validate whether group discussions are segregated by topic and can be accessed through appropriate filters.

Test Suite: Events and Groups

Test Priority: Medium

Preconditions:

- * User is logged in.
- * User is a member of a group on the Social Media Platform.

Test Data:

- * User with valid login credentials.

Test Steps:

1. Navigate to the Groups section and select the desired group.
2. Locate the "Discussions" tab on the group's page.
3. Verify the existence of different topic-wise threads in the discussions.
4. Search for a topic using the search bar.
5. Validate whether it returns correct results.

Postconditions:

- * User can view discussions segregated by topic.
- * User can search discussions using keywords successfully.

Expected Result:

* Group discussions are organized by topic and can be accessed through filters, making it easier for users to find relevant conversations.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, User Interface Testing

Test Case Approach: Positive

Test Case ID: TC-EG-004

Test Case Title: Verify that invalid RSVP information is handled appropriately

Test Case Description: Validate the platform's ability to handle incorrect RSVP inputs and display appropriate error messages.

Test Suite: Events and Groups

Test Priority: Medium

Preconditions:

- * User is logged in.
- * User has access to the Social Media Platform.
- * Event exists on the platform.

Test Data:

- * User with valid login credentials.

Test Steps:

1. Navigate to the Events section on the Social Media Platform.
2. Search for an event by date, category, or location.
3. Select the desired event from the list of results.
4. Click the "RSVP" button on the event's page.
5. Enter incorrect information on the confirmation dialog (e.g., invalid date or time).
6. Submit the RSVP.

Postconditions:

- * User cannot complete the RSVP process with incorrect inputs.

Expected Result:

- * The platform returns an appropriate error message indicating that the input is invalid.

Severity: Major

Priority: Medium

Type of Testing: Negative Testing

Test Case Approach: Negative

Test Case ID: TC-EG-005

Test Case Title: Verify that users can create events within a group

Test Case Description: Validate the ability for users to create and manage events within their interest-based groups.

Test Suite: Events and Groups

Test Priority: High

Preconditions:

- * User is logged in.
- * User is a member of a group on the Social Media Platform.

Test Data:

- * User with valid login credentials.
- * Event details including date, time, location, and description.

Test Steps:

1. Navigate to the Groups section and select the desired group.
2. Locate the "Create Event" button on the group's page.
3. Enter the event details into the provided form (date, time, location, and description).
4. Click the "Create Event" button.
5. Verify that the new event is present in the group's events list.

6. Verify that the user can edit and delete the newly created event.

Postconditions:

* Event has been created within the group.

* User can manage the event.

Expected Result:

* Users have successfully created and managed an event within their interest-based group.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 14

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

Monetization Options

Feature description -

Monetization options allow users to earn money through various means on the platform. This may include advertising revenue, sponsored content partnerships, affiliate marketing programs, subscriptions, virtual gifts, or donations from followers, providing creators and influencers with opportunities to monetize their content

and audience.

Userstory title -

As a content creator, I want to access monetization options so that I can earn revenue from my content.

Acceptance criteria -

['Users can apply for monetization options.', 'Monetization includes advertising revenue and sponsored content.', 'Users can receive payments through the platform.', 'Detailed earnings reports are available.', 'Monetization eligibility requirements are clearly stated.']

Generated Test Cases:

Test Case ID: MON-001

Test Case Title: Verify that users can apply for monetization options

Test Case Description: This test case verifies that a registered user can successfully apply for monetization options through the platform's settings.

Test Suite: Monetization

Test Priority: High

Preconditions:

1. User is logged into their account.
2. User has access to the account settings page.

Test Data: No test data needed

Test Steps:

1. Navigate to the account settings page.
2. Click on the "Monetization" tab.
3. Select the "Apply Now" button.
4. Complete the monetization application form with valid information.
5. Submit the monetization application.

Postconditions:

1. User receives a confirmation message that their application has been submitted.

2. The user's account status updates to "Pending Monetization Approval."

Expected Result: The test case passes if the user can successfully submit a monetization application and their account status is updated accordingly.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: MON-002

Test Case Title: Verify that monetization includes advertising revenue and sponsored content

Test Case Description: This test case verifies that a user's monetization account displays advertising revenue and sponsored content as sources of income.

Test Suite: Monetization

Test Priority: High

Preconditions:

1. User has an approved monetization account.
2. User has advertising revenue and sponsored content in their account.

Test Data: No test data needed

Test Steps:

1. Log in to the user account.
2. Navigate to the monetization dashboard.
3. Check the "Revenue Sources" section for "Advertising" and "Sponsored Content" as options.
4. Verify the amounts earned from each revenue source.

Postconditions:

1. User sees accurate revenue information for both advertising and sponsored content.

Expected Result: The test case passes if the user's monetization dashboard displays advertising revenue and sponsored content as monetization sources with accurate earnings information.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: MON-003

Test Case Title: Verify that users can receive payments through the platform

Test Case Description: This test case verifies that a user with an approved monetization account can successfully receive payments through the platform.

Test Suite: Monetization

Test Priority: High

Preconditions:

1. User has an approved monetization account.
2. User has earnings available for payout.

Test Data: No test data needed

Test Steps:

1. Log in to the user account.
2. Navigate to the monetization dashboard.
3. Click on the "Request Payout" button.
4. Provide payout information (e.g., bank account, PayPal, etc.).
5. Confirm the payout request.

Postconditions:

1. User receives a confirmation message that their payout request has been initiated.
2. User's account balance updates accordingly.

Expected Result: The test case passes if the user can successfully initiate a payout request and their account balance updates accordingly.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: MON-004

Test Case Title: Verify that detailed earnings reports are available

Test Case Description: This test case verifies that a user with an approved monetization account can view detailed earnings reports for their content.

Test Suite: Monetization

Test Priority: Medium

Preconditions:

1. User has an approved monetization account.
2. User has earnings available for reporting.

Test Data: No test data needed

Test Steps:

1. Log in to the user account.
2. Navigate to the monetization dashboard.
3. Click on the "Earnings Report" tab.
4. Select a date range for the report.
5. View the detailed earnings report for the selected date range.

Postconditions:

1. User sees a detailed earnings report with various data points (e.g., revenue sources, earnings by content, etc.).

Expected Result: The test case passes if the user can view a detailed earnings report with accurate data points for the selected date range.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: MON-005

Test Case Title: Verify that monetization eligibility requirements are clearly stated

Test Case Description: This test case verifies that the monetization eligibility requirements are clearly stated and accessible for users.

Test Suite: Monetization

Test Priority: Low

Preconditions: N/A

Test Data: No test data needed

Test Steps:

1. Navigate to the platform's public website.
2. Locate the monetization information or FAQ page.
3. Read through the monetization eligibility requirements.

Postconditions: N/A

Expected Result: The test case passes if the monetization eligibility requirements are easily accessible and clearly stated for users.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 15

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

Explore or Discover Tab

Feature description -

The explore or discover tab curates trending or popular content from across the platform, helping users discover new accounts, topics, and trends. The tab may feature personalized recommendations based on users' interests, engagement history, and social connections, facilitating content discovery and exploration.

Userstory title -

As a user, I want to explore trending content so that I can discover new and popular posts, accounts, and topics.

Acceptance criteria -

['The explore tab shows trending and popular content.', 'Content recommendations are personalized based on user interests.', 'Users can filter content by category or topic.', 'The tab updates regularly with new content.', 'Users can save or follow content from the explore tab.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that the explore tab shows trending and popular content.

Test Case Description: As a user, I want to ensure that the explore tab displays the most popular and trending content on the social media platform.

Test Suite: Explore Tab Functionality

Test Priority: High

Preconditions:

- User is logged in
- User is on the explore tab

Test Data: No test data needed

Test Steps:

1. Log in to the social media platform.
2. Navigate to the explore tab.
3. Observe the content displayed on the explore tab.

Postconditions: N/A

Expected Result: The explore tab displays popular and trending content from various accounts and topics.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that content recommendations are personalized based on user interests.

Test Case Description: As a user, I want to make sure that the content recommendations on the explore tab are tailored to my interests, engagement history, and social connections.

Test Suite: Content Personalization

Test Priority: High

Preconditions:

- User is logged in
- User has a history of engagement on the platform

Test Data: User engagement history

Test Steps:

1. Log in to the social media platform.

2. Engage with various posts, accounts, and topics.
3. Navigate to the explore tab.
4. Observe the content recommendations.

Postconditions: N/A

Expected Result: The explore tab presents personalized content recommendations based on user interests, engagement history, and social connections.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users can filter content by category or topic.

Test Case Description: As a user, I want to be able to filter the content on the explore tab by category or topic to discover new and relevant content.

Test Suite: Content Filtering

Test Priority: Medium

Preconditions:

- User is logged in
- User is on the explore tab

Test Data: Various categories and topics

Test Steps:

1. Log in to the social media platform.
2. Navigate to the explore tab.
3. Use the filter options to select a category or topic.
4. Observe the filtered content.

Postconditions: N/A

Expected Result: The explore tab displays filtered content by the chosen category or topic.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that the explore tab updates regularly with new content.

Test Case Description: As a user, I want to ensure that the explore tab updates frequently with new and trending content to keep my discovery experience fresh.

Test Suite: Content Update

Test Priority: Medium

Preconditions:

- User is logged in
- User is on the explore tab

Test Data: Timestamps of content updates

Test Steps:

1. Log in to the social media platform.
2. Navigate to the explore tab.
3. Observe the timestamps of the content displayed.
4. Wait for a predefined period (e.g., 1 hour).
5. Navigate back to the explore tab and compare the timestamps of the content displayed before and after the waiting period.

Postconditions: N/A

Expected Result: The explore tab updates with new content regularly.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that users can save or follow content from the explore tab.

Test Case Description: As a user, I want to be able to save or follow content from the explore tab so that I can easily access it later or stay updated on new content.

Test Suite: Content Save and Follow

Test Priority: Low

Preconditions:

- User is logged in
- User is on the explore tab

Test Data: Specific content to save or follow

Test Steps:

1. Log in to the social media platform.
2. Navigate to the explore tab.
3. Choose a piece of content to save or follow.
4. Use the save or follow function on the chosen content.
5. Check if the content has been saved or the account has been followed.

Postconditions: N/A

Expected Result: Users can save or follow content from the explore tab.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 16

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

Accessibility Features

Feature description -

Accessibility features ensure that the platform is inclusive and accessible to users with disabilities. This may include features such as alt text for images, screen reader compatibility, keyboard shortcuts, color contrast options, and adjustable font sizes, enhancing usability and accessibility for all users.

Userstory title -

As a user with disabilities, I want accessible features so that I can use the platform comfortably.

Acceptance criteria -

['The platform provides alt text for images.', 'Screen reader compatibility is available.', 'Keyboard shortcuts can be used for navigation.', 'Color contrast and font size options are provided.', 'Accessibility features are clearly documented.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC001_Accessibility_AltText

Test Case Title: Verify that the platform provides alt text for images

Test Case Description: This test case verifies if the social media platform provides descriptive alt text for images, enabling visually impaired users to understand the content through screen readers.

Test Suite: Accessibility Features

Test Priority: High

Preconditions:

- User is logged in and on the news feed page
- An image post is available on the news feed

Test Data: No test data needed

Test Steps:

1. Log in as a user
2. Navigate to the news feed
3. Identify an image post
4. Right-click on the image and inspect the element
5. Verify if the tag contains an "alt" attribute with a descriptive value

Postconditions:

- User remains logged in
- News feed page is still displayed

Expected Result: The test case passes if each image on the news feed has an "alt" attribute with an appropriate descriptive value.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Usability Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC002_Accessibility_ScreenReader

Test Case Title: Verify that the platform is compatible with screen readers

Test Case Description: This test case checks if the social media platform is compatible with popular screen readers, allowing visually impaired users to navigate the platform and consume content effectively.

Test Suite: Accessibility Features

Test Priority: High

Preconditions:

- User is logged in and on the news feed page
- A screen reader (e.g., JAWS, NVDA) is installed and enabled

Test Data: No test data needed

Test Steps:

1. Log in as a user
2. Navigate to the news feed
3. Enable the screen reader
4. Navigate through the page using the screen reader
5. Verify if the screen reader accurately reads out the UI elements, posts, and other content

Postconditions:

- User remains logged in
- News feed page is still displayed
- Screen reader is enabled

Expected Result: The test case passes if the screen reader accurately reads out all UI elements, post content, and other relevant information.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Usability Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC003_Accessibility_KeyboardNavigation

Test Case Title: Verify that the platform supports keyboard shortcuts for navigation

Test Case Description: This test case validates if the social media platform allows users to navigate through UI elements and perform actions using keyboard shortcuts.

Test Suite: Accessibility Features

Test Priority: Medium

Preconditions:

- User is logged in and on the news feed page

Test Data: No test data needed

Test Steps:

1. Log in as a user
2. Navigate to the news feed
3. Press the "Tab" key to cycle through UI elements
4. Verify if the focus order is logical and sequential
5. Perform actions (e.g., like/comment/share) using keyboard shortcuts

Postconditions:

- User remains logged in
- News feed page is still displayed

Expected Result: The test case passes if keyboard shortcuts can be used to navigate through all UI elements and perform actions on posts.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Usability Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC004_Accessibility_ColorContrast

Test Case Title: Verify that the platform provides color contrast and font size options

Test Case Description: This test case checks if the social media platform supports customizable color contrast and font size options, allowing users with color blindness or visual impairments to use the platform comfortably.

Test Suite: Accessibility Features

Test Priority: Medium

Preconditions:

- User is logged in

Test Data: No test data needed

Test Steps:

1. Log in as a user
2. Access the accessibility settings page
3. Change the color scheme and font size
4. Verify if the new color scheme and font size are applied consistently across the platform

Postconditions:

- User remains logged in
- Accessibility settings page is displayed

Expected Result: The test case passes if the color contrast and font size can be customized and the new settings are applied consistently across the platform.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Usability Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC005_Accessibility_Documentation

Test Case Title: Verify that accessibility features are clearly documented

Test Case Description: This test case validates if the social media platform's documentation includes clear and comprehensive instructions for enabling and using the accessibility features.

Test Suite: Documentation

Test Priority: Low

Preconditions:

- Documentation is available online

Test Data: No test data needed

Test Steps:

1. Access the platform's official documentation
2. Look for a dedicated section on accessibility features
3. Verify if the section includes detailed instructions for enabling and using the features (e.g., alt text, screen reader compatibility, keyboard shortcuts, color contrast options, and adjustable font sizes)

Postconditions:

- Documentation is still available online

Expected Result: The test case passes if the documentation provides clear and comprehensive instructions for enabling and using accessibility features.

Severity: Major

Priority: Low

Type of Testing: Documentation Review, Usability Testing

Test Case Approach: Positive

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

Safety and Moderation Tools

Feature description -

Safety and moderation tools are designed to protect users from harmful or inappropriate content and interactions on the platform. This may include features such as content moderation algorithms, reporting tools, blocking capabilities, comment filters, and community guidelines enforcement, fostering a safer and more positive online environment.

Userstory title -

As a user, I want safety and moderation tools so that I can have a safe and positive experience on the platform.

Acceptance criteria -

['Content moderation algorithms filter inappropriate content.', 'Users can report harmful or inappropriate content.', 'Blocking capabilities are available to users.', 'Comment filters prevent offensive language.', 'Community guidelines are enforced consistently.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that content moderation algorithms filter inappropriate content

Test Suite: Safety and Moderation Tools

Test Priority: High

Preconditions:

- User is logged in
- User has created a post with text

Test Data: A post containing offensive language

Test Steps:

1. Log in as a user
2. Create a new post with offensive language
3. Publish the post
4. Verify if the post is visible on the user's timeline
5. Check if the post is visible in the community feed

Postconditions:

- Offensive post should not be visible in either location

Expected Result: The offensive post is not displayed on the user's timeline or in the community feed, demonstrating that content moderation algorithms effectively filter inappropriate content.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Negative

Test Case ID: TC002

Test Case Title: Verify that users can report harmful or inappropriate content

Test Suite: Safety and Moderation Tools

Test Priority: High

Preconditions:

- User is logged in
- User has access to the reporting feature

Test Data: A post containing hate speech

Test Steps:

1. Log in as a user
2. Navigate to a post containing hate speech
3. Click on the report button
4. Select the appropriate reason for reporting from the dropdown menu
5. Confirm the report

Postconditions:

- The reported post should be under review

Expected Result: The user can successfully report a post containing hate speech, which is then marked for review by moderators.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that blocking capabilities are available to users

Test Suite: Safety and Moderation Tools

Test Priority: High

Preconditions:

- User A is logged in
- User B is logged in
- User A and User B are connected

Test Data: None

Test Steps:

1. Log in as User A
2. Navigate to User B's profile
3. Click on the block button

4. Confirm the block action

Postconditions:

- User A should not see posts from User B in their feed
- User A should not be able to send messages to User B

Expected Result: User A can successfully block User B, resulting in User A no longer seeing User B's posts in their feed or being able to send messages to User B.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that comment filters prevent offensive language

Test Suite: Safety and Moderation Tools

Test Priority: High

Preconditions:

- User is logged in
- User has access to the commenting feature

Test Data: A comment containing offensive language

Test Steps:

1. Log in as a user
2. Create a new post
3. In the comment section, enter a comment with offensive language
4. Attempt to post the comment

Postconditions:

- Offensive comment should not be posted

Expected Result: The offensive comment is not posted, showing that comment filters effectively prevent the

use of offensive language.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Negative

Test Case ID: TC005

Test Case Title: Verify that community guidelines are enforced consistently

Test Suite: Safety and Moderation Tools

Test Priority: High

Preconditions:

- Multiple users are logged in
- Users have access to the reporting feature

Test Data: Various posts, some containing hate speech, and others not

Test Steps:

1. Log in as multiple users
2. Have users report posts containing hate speech
3. Wait for a moderator to review the reported posts

Postconditions:

- Posts containing hate speech should be removed

Expected Result: Moderators consistently enforce community guidelines by removing posts containing hate speech, demonstrating a safe and positive platform experience for users.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive (but requires human intervention)

Data - 18

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

Multi-Language Support

Feature description -

Multi-language support allows the platform to cater to users from diverse linguistic backgrounds by offering interface localization and content translation features. Users can select their preferred language for the interface, and posts or messages can be translated into different languages, enhancing accessibility and user experience across global audiences.

Userstory title -

As a user, I want multi-language support so that I can use the platform in my preferred language.

Acceptance criteria -

['Users can select their preferred interface language.', 'The platform offers content translation features.', 'Posts and messages can be translated into different languages.', 'Language preferences are saved and applied consistently.', 'The platform supports a wide range of languages.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that users can select their preferred interface language

Test Case Description: This test case aims to verify that the social media platform allows users to choose their preferred language for the interface, ensuring a localized user experience.

Test Suite: User Interface Localization

Test Priority: High

Preconditions:

1. User is logged in to the social media platform.
2. User has access to the language settings.

Test Data: No test data needed

Test Steps:

1. Log in to the social media platform.
2. Access the language settings from the user profile.
3. Select a language other than the default one.
4. Save and apply the selected language preferences.
5. Navigate to other pages to validate if the language change is consistent.

Postconditions:

1. Language preferences are saved and applied consistently across the platform.
2. Interface elements and features are displayed in the selected language.

Expected Result: Language preferences are saved and applied consistently across the platform, with interface elements and features displayed in the selected language.

Severity: Major

Priority: High

Type of Testing: Functional Testing, User Interface Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that the platform offers content translation features

Test Case Description: This test case aims to validate that the social media platform includes a content

translation feature, enabling users to translate posts and messages into their preferred language.

Test Suite: Content Translation

Test Priority: High

Preconditions:

1. User is logged in to the social media platform.
2. User has access to the translation feature.

Test Data: A post or message in a language other than the user's preferred language.

Test Steps:

1. Log in to the social media platform.
2. Locate a post or message in a language other than the user's preferred language.
3. Access the translation feature from the post or message.
4. Select the user's preferred language for the translation.
5. Verify the translated content's accuracy.

Postconditions:

1. Post or message is translated into the user's preferred language.
2. Translated content is displayed correctly and intelligibly.

Expected Result: Post or message is translated into the user's preferred language, displayed correctly and intelligibly, with accuracy in meaning and context.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Content Translation Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that the platform supports a wide range of languages

Test Case Description: This test case aims to validate that the social media platform supports a comprehensive list of languages for interface localization and content translation.

Test Suite: Language Support

Test Priority: Medium

Preconditions:

1. User has access to the language settings.

Test Data: List of supported languages provided by the project requirements.

Test Steps:

1. Access the language settings from the user profile.
2. Verify that the list of supported languages includes all the languages specified in the project requirements.

Postconditions:

1. All specified languages are supported for interface localization and content translation.

Expected Result: All specified languages are supported for interface localization and content translation, ensuring global accessibility and user experience.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Language Support Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that language preferences are saved and applied consistently

Test Case Description: This test case aims to validate that the social media platform saves and applies language preferences consistently for the user interface and content throughout the platform.

Test Suite: Language Preference Saving and Application

Test Priority: Medium

Preconditions:

1. User is logged in to the social media platform.
2. User has set a preferred language.

Test Data: No test data needed

Test Steps:

1. Log in to the social media platform.
2. Navigate to different pages and features within the platform.
3. Verify that the user interface is displayed in the preferred language.
4. Locate content in a language other than the user's preferred language.
5. Access the translation feature and translate the content.
6. Verify that the translated content is displayed in the preferred language.

Postconditions:

1. Language preferences are saved and applied consistently across the platform.
2. Translated content is displayed correctly and intelligibly in the preferred language.

Expected Result: Language preferences are saved and applied consistently across the platform, with interface elements and translated content displayed correctly and intelligibly in the user's preferred language.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that the platform handles language switching gracefully without crashes

Test Case Description: This test case aims to validate that the social media platform handles switching between languages seamlessly without crashes, maintaining a smooth user experience.

Test Suite: Language Switching

Test Priority: Low

Preconditions:

1. User is logged in to the social media platform.
2. User has access to the language settings.

Test Data: No test data needed

Test Steps:

- 1. Log in to the social media platform.
- 2. Change the language preference to another supported language.
- 3. Navigate to different pages and features within the platform.
- 4. Repeat steps 2 and 3 for multiple language switches.

Postconditions:

- 1. The platform maintains a smooth user experience while switching between languages.
- 2. No crashes or severe errors occur during the language switching process.

Expected Result: The platform handles language switching seamlessly without crashes, maintaining a smooth user experience.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing, Regression Testing, User Experience Testing

Test Case Approach: Positive

Data - 19

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

Cross-Platform Integration

Feature description -

Cross-platform integration allows users to connect and share content across different social media platforms and external services. This includes features like single sign-on (SSO) capabilities, cross-posting of content, integration with external media sources, and the ability to share platform content on other social networks, enhancing user convenience and content reach.

Userstory title -

As a user, I want cross-platform integration so that I can connect and share content across multiple platforms.

Acceptance criteria -

['Users can link their accounts with other social media platforms.', 'Content can be cross-posted to other platforms.', 'Users can sign in using single sign-on (SSO).', 'Integration with external media sources is available.', 'Content sharing across platforms is seamless.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC001_CrossPlatformIntegration_SSO

Test Case Title: Verify that users can sign in using single sign-on (SSO) functionality.

Test Case Description: This test case verifies the proper functioning of the single sign-on (SSO) feature for user authentication.

Test Suite: Authentication Module

Test Priority: High

Preconditions:

- The user is registered on the Social Media Platform.
- The user has an account on a supported third-party authentication service (e.g. Google, Facebook).

Test Data: User credentials for the supported third-party authentication service.

Test Steps:

1. Navigate to the Social Media Platform login page.
2. Select the option to sign in using the third-party authentication service.

3. Enter the user credentials in the third-party authentication service prompt.
4. Grant permission for the Social Media Platform to access the third-party authentication service.
5. Verify that the user is logged into the Social Media Platform.

Postconditions:

- The user is logged into the Social Media Platform.
- The user's third-party authentication service account is linked to their Social Media Platform account.

Expected Result: The user can sign in using single sign-on (SSO) functionality with a supported third-party authentication service without any issues.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC002_CrossPlatformIntegration_LinkAccounts

Test Case Title: Verify that users can link their accounts with other social media platforms.

Test Case Description: This test case checks if users can successfully link their Social Media Platform accounts with other social media platform accounts.

Test Suite: Integration Module

Test Priority: Medium

Preconditions:

- The user is registered on the Social Media Platform.
- The user has an account on the external social media platform.

Test Data: User credentials for the Social Media Platform and the external social media platform.

Test Steps:

1. Navigate to the Social Media Platform account settings.
2. Select the option to link an external social media platform account.

3. Enter the user credentials for the external social media platform.
4. Verify that the user is redirected back to the Social Media Platform.
5. Confirm that the external social media platform account is now linked to the user's Social Media Platform account.

Postconditions:

- The external social media platform account is linked to the user's Social Media Platform account.

Expected Result: Users can link their Social Media Platform accounts with other social media platform accounts without any issues.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC003_CrossPlatformIntegration_CrossPosting

Test Case Title: Verify that users can cross-post content from the Social Media Platform to other platforms.

Test Case Description: This test case verifies users' ability to cross-post content from the Social Media Platform to other platforms.

Test Suite: Content Sharing Module

Test Priority: High

Preconditions:

- The user is registered on the Social Media Platform.
- The user has linked at least one external social media platform account.

Test Data: A new post, photo, or video.

Test Steps:

1. Create a new post, photo, or video on the Social Media Platform.
2. Select the option to cross-post the content.

3. Choose at least one external social media platform account to share the content.
4. Verify that the content has been successfully shared on the chosen external social media platforms.

Postconditions:

- The content has been shared on the external social media platforms.

Expected Result: Users can cross-post content from the Social Media Platform to other platforms without any issues.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC004_CrossPlatformIntegration_IntegrationExternalMedia

Test Case Title: Verify that integration with external media sources is available.

Test Case Description: This test case checks if external media sources (such as YouTube and Spotify) can be integrated and shared on the Social Media Platform.

Test Suite: External Media Integration Module

Test Priority: Medium

Preconditions:

- The user is registered on the Social Media Platform.
- The user has access to at least one external media source.

Test Data: A link from an external media source.

Test Steps:

1. Navigate to the Social Media Platform's post creation page.
2. Paste a link from the external media source.
3. Verify that the link is correctly parsed and displayed in the post preview.
4. Publish the post.

5. Confirm that the external media source is accessible within the post.

Postconditions:

- The external media source is accessible within the post.

Expected Result: Integration with external media sources works correctly.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC005_CrossPlatformIntegration_SeamlessSharing

Test Case Title: Verify that content sharing across platforms is seamless.

Test Case Description: This test case verifies that users can share content across different social media platforms seamlessly.

Test Suite: Content Sharing Module

Test Priority: High

Preconditions:

- The user is registered on the Social Media Platform.
- The user has linked at least one external social media platform account.

Test Data: A new post.

Test Steps:

1. Create a new post on the Social Media Platform.
2. Share the content on an external social media platform.
3. Verify that the content has been correctly shared on the external social media platform.
4. Make an edit to the original post on the Social Media Platform.
5. Verify that the edit is reflected in the shared post on the external social media platform.

Postconditions:

- Content has been shared and updated correctly across platforms.

Expected Result: Content sharing across platforms is seamless and updates are reflected in real-time.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 20

Project name -

Social Media Platform

Project description -

Develop a comprehensive social media platform where users can create profiles, connect with friends, share posts, photos, and videos, engage with content through likes, comments, and shares, and explore trending topics and communities. The platform should include features like news feed, messaging, user profiles, privacy settings, notifications, and analytics.

Feature name -

Gamification and Rewards

Feature description -

Gamification and rewards enhance user engagement by incorporating game-like elements such as points, badges, leaderboards, and challenges. Users can earn rewards for their activities, achievements, and contributions on the platform, fostering a sense of accomplishment and encouraging continued interaction and participation.

Userstory title -

As a user, I want gamification and rewards so that I can earn recognition and incentives for my activities.

Acceptance criteria -

['Users can earn points and badges for their activities.', 'Leaderboards display top users based on points or achievements.', 'Challenges are available for users to complete and earn rewards.', 'Rewards can include virtual goods, discounts, or exclusive content.', 'Users can view their progress and achievements in a dedicated section.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that users can earn points and badges for their activities

Test Case Description: Validate the gamification system by performing various activities, such as posting, commenting, and sharing content, and checking if the user's point balance and badge collection increase accordingly.

Test Suite: Gamification and Rewards

Test Priority: High

Preconditions:

- User must be logged in
- User must have access to posting, commenting, and sharing features

Test Data: User activity data, including posts, comments, and shares

Test Steps:

1. Log in as a registered user
2. Perform an activity, such as posting a status update
3. Verify if the user's point balance has increased
4. Perform another activity, such as commenting on a friend's post
5. Verify if the user's point balance has increased and if a new badge has been awarded
6. Perform a third activity, such as sharing a post
7. Verify if the user's point balance has increased and if a new badge has been awarded

Postconditions:

- User's point balance and badge collection reflect the performed activities

Expected Result: User can earn points and badges for their activities

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that leaderboards display top users based on points or achievements

Test Case Description: Check the leaderboard functionality by comparing the displayed users' points or achievements with the actual user data.

Test Suite: Gamification and Rewards

Test Priority: High

Preconditions:

- User must be logged in
- User must have access to leaderboard feature

Test Data: User activity data, including points and badges

Test Steps:

1. Log in as a registered user
2. Access the leaderboard feature
3. Verify if the displayed users' points or badges match the actual user data
4. Check if the top users are correctly displayed based on points or achievements
5. Verify if filtering options for points or badges work as expected

Postconditions:

- Leaderboard displays accurate user data and ranks users correctly

Expected Result: Leaderboard displays top users based on points or achievements

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that challenges are available for users to complete and earn rewards

Test Case Description: Validate challenge availability, completion, and reward assignment.

Test Suite: Gamification and Rewards

Test Priority: Medium

Preconditions:

- User must be logged in
- User must have access to challenge feature

Test Data: Available challenges and associated rewards

Test Steps:

1. Log in as a registered user
2. Access the challenge feature
3. Verify if challenges are available and display correct information
4. Select a challenge and complete the required task
5. Verify if the challenge status updates to "completed"
6. Check if the associated reward is granted to the user

Postconditions:

- Challenges are available and can be completed with assigned rewards

Expected Result: Users can complete challenges and earn rewards

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that rewards can include virtual goods, discounts, or exclusive content

Test Case Description: Ensure that different types of rewards are granted to users based on their activities or achievements.

Test Suite: Gamification and Rewards

Test Priority: Medium

Preconditions:

- User must be logged in
- User must have access to rewards feature
- User must have completed eligible activities or achievements

Test Data: Eligible activities or achievements and corresponding rewards

Test Steps:

1. Log in as a registered user
2. Access the rewards feature
3. Verify that virtual goods, discounts, and exclusive content are available
4. Complete eligible activities or achievements
5. Verify if the granted rewards match the completed activities or achievements

Postconditions:

- Users receive the correct rewards for their activities or achievements

Expected Result: Rewards include virtual goods, discounts, or exclusive content

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that users can view their progress and achievements in a dedicated section

Test Case Description: Validate that users can access and view their progress, including points and badges, in a dedicated section of the platform.

Test Suite: Gamification and Rewards

Test Priority: Low

Preconditions:

- User must be logged in
- User must have access to user profile

Test Data: User activity data, including points and badges

Test Steps:

1. Log in as a registered user
2. Access the user profile section
3. Verify if the progress and achievements section is available
4. Check if point balance and badge collection are displayed accurately
5. Verify if the section includes information about completed challenges and earned rewards

Postconditions:

- Users can view their progress and achievements in a dedicated section

Expected Result: Progress and achievements are displayed accurately in the user profile

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 21

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Creation

Feature description -

Users can create tasks by providing detailed information such as the task title, description, due date, priority level, and assigning it to specific team members. This feature serves as the foundation for organizing and managing tasks within the tool.

Userstory title -

As a user, I want to create tasks so that I can organize and manage my work effectively.

Acceptance criteria -

['Users can enter a task title, description, due date, and priority level.', 'Tasks can be assigned to specific team members.', 'The system should save and display created tasks.', 'Users can edit task details after creation.', 'The system should validate required fields (e.g., task title).']

Generated Test Cases:

Test Case 1:

Test Case ID: TST001

Test Case Title: Verify that users can enter a task title

Test Case Description: This test case aims to ensure that users can input a task title in the designated field during the task creation process.

Test Suite: Task Creation

Test Priority: High

Preconditions:

- User is logged in
- User navigates to the "Create Task" page

Test Data: Any valid alphanumeric string

Test Steps:

1. Enter "Test Task Title" in the task title field
2. Click the "Create Task" button

Postconditions:

- The system saves the task with the given title

Expected Result: The system displays the new task with the entered title.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TST002

Test Case Title: Verify that users can enter a task description

Test Case Description: This test case validates if users can provide a detailed task description when creating a new task.

Test Suite: Task Creation

Test Priority: Medium

Preconditions:

- User is logged in
- User navigates to the "Create Task" page

Test Data: Any valid text string up to the maximum character limit

Test Steps:

1. Enter "This is a test task description." in the task description field
2. Click the "Create Task" button

Postconditions:

- The system saves the task with the given description

Expected Result: The system displays the new task with the entered description.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TST003

Test Case Title: Verify that users can assign tasks to specific team members

Test Case Description: This test case checks the ability for users to assign tasks to other team members during the task creation process.

Test Suite: Task Creation

Test Priority: High

Preconditions:

- User is logged in
- User navigates to the "Create Task" page
- Other team members are registered in the system

Test Data: Another team member's username

Test Steps:

1. Enter a task title and description
2. Select a team member from the "Assign to" dropdown
3. Click the "Create Task" button

Postconditions:

- The system saves the task and assigns it to the selected team member

Expected Result: The system displays the new task assigned to the chosen team member.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TST004

Test Case Title: Verify that the system saves and displays created tasks

Test Case Description: This test case aims to confirm that the system saves and correctly displays the newly created tasks.

Test Suite: Task Creation

Test Priority: High

Preconditions:

- User is logged in

Test Steps:

1. Enter a task title, description, and assign it to a team member
2. Click the "Create Task" button

Postconditions:

- The system saves the task

Expected Result: The system shows the new task in the task list for the user and the assigned team member.

Severity: Blocker

Priority: High

Type of Testing: System Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TST005

Test Case Title: Verify that required fields are validated

Test Case Description: This test case tests the system's ability to validate required fields during the task creation process.

Test Suite: Task Creation

Test Priority: Medium

Preconditions:

- User is logged in
- User navigates to the "Create Task" page

Test Data: Leaving the task title field blank

Test Steps:

1. Leave the task title field empty
2. Click the "Create Task" button

Postconditions: None

Expected Result: An error message should be displayed, prompting the user to input a task title.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Negative

Data - 22

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Assignment

Feature description -

Tasks can be assigned to specific team members, delegating responsibility and ensuring clarity about who is accountable for each task. Assignees receive notifications about their assigned tasks, helping them stay informed and focused on their responsibilities.

Userstory title -

As a user, I want to assign tasks to specific team members so that responsibilities are clear and tasks are managed effectively.

Acceptance criteria -

['Users can select team members to assign tasks.', 'Assignees receive notifications about their tasks.', 'The system should display assigned tasks in the assignee's task list.', 'Users can reassign tasks to different team members.', 'Task assignment history should be tracked.']

Generated Test Cases:

Test Case 1:

Test Case ID: TMT-001

Test Case Title: Verify that users can select team members to assign tasks

Test Case Description: As a user, I want to be able to assign tasks to specific team members from a list of available members.

Test Suite: Task Assignment

Test Priority: High

Preconditions:

- User is logged in
- At least one other team member exists in the system

Test Data: A new task with a title and description

Test Steps:

1. Create a new task
2. Enter a title and description for the task
3. Click the "Assign" button
4. Select a team member from the dropdown list
5. Save the task

Postconditions:

- Assigned team member should receive a notification
- Task should appear in the assignee's task list

Expected Result: The task is assigned to the selected team member, and the system displays the assigned task in the assignee's task list.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TMT-002

Test Case Title: Verify that assignees receive notifications about their tasks

Test Case Description: Assignees should receive notifications when they are assigned a new task, helping them stay informed about their responsibilities.

Test Suite: Task Assignment Notification

Test Priority: Medium

Preconditions:

- User is logged in
- Another team member is available in the system

Test Data: A new task with a title, description, and assignee

Test Steps:

1. Create a new task
2. Enter a title and description for the task
3. Assign the task to a team member
4. Save the task

Postconditions:

- Assigned team member should receive a notification

Expected Result: The assignee receives a notification about the newly assigned task.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TMT-003

Test Case Title: Verify that users can reassign tasks to different team members

Test Case Description: Users should have the ability to reassign tasks from one team member to another, ensuring that responsibilities can be adjusted as needed.

Test Suite: Task Reassignment

Test Priority: High

Preconditions:

- User is logged in
- A task exists, previously assigned to a team member

Test Data: A task assigned to a team member

Test Steps:

1. Navigate to the task
2. Select the current assignee's name
3. Choose a new team member from the dropdown list
4. Save the reassignment

Postconditions:

- Original assignee should no longer have the task
- New assignee should have the task

Expected Result: The task is reassigned from the original assignee to the newly selected team member, and the system updates both users' task lists accordingly.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TMT-004

Test Case Title: Verify that the system tracks task assignment history

Test Case Description: Tasks should have an assignment history, which includes information about who has been assigned to the task and when.

Test Suite: Task Assignment History

Test Priority: Medium

Preconditions:

- User is logged in
- A task exists, previously assigned to a team member

Test Data: A task with an assignee

Test Steps:

1. Navigate to the task
2. View the assignment history

Postconditions:

- Assignment history is visible

Expected Result: The system displays a clear history of task assignments, including information about when and to whom the task was assigned.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TMT-005

Test Case Title: Verify that the system displays assigned tasks in the assignee's task list

Test Case Description: Once a task is assigned to a team member, the task should appear in their task list as an assigned task.

Test Suite: Task List Display

Test Priority: High

Preconditions:

- User is logged in as the assignee
- A task exists, previously assigned to the logged-in user

Test Data: A task assigned to the logged-in user

Test Steps:

1. Log in as the assignee
2. Navigate to their task list

Expected Result: The assigned task is displayed in the user's task list, with relevant information (e.g., title, description, due date).

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 23

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Deadline Reminders

Feature description -

Users receive reminders and notifications for upcoming task deadlines, ensuring that tasks are completed on time and preventing delays. Reminders can be customized based on users' preferences and urgency levels.

Userstory title -

As a user, I want to receive reminders for upcoming task deadlines so that I can complete tasks on time.

Acceptance criteria -

['Users can set reminders for tasks based on deadlines.', 'Reminders can be customized by urgency level.', 'The system sends notifications for upcoming deadlines.', 'Users can choose the delivery method for reminders (e.g., email, in-app).', 'Reminder settings can be adjusted or disabled.']

Generated Test Cases:

Test Case 1:

Test Case ID: TCTM-TD-001

Test Case Title: Verify that users can set reminders for tasks based on deadlines.

Test Case Description: As a user, I want to set a reminder for a task so that I am notified before the deadline.

Test Suite: Task Management

Test Priority: High

Preconditions:

1. User is logged in.
2. User has created a task.
3. User has not set a reminder for the task.

Test Data: Selected task with a deadline.

Test Steps:

1. Navigate to the task list.
2. Select a task with a deadline.
3. Click on the "Set Reminder" button.
4. Enter a reminder time, e.g. 1 day before the deadline.

5. Confirm the reminder setup.

Postconditions:

1. Reminder time is saved for the selected task.

2. User is notified 1 day before the deadline.

Expected Result: The system creates a reminder for the user based on the set deadline and reminder time.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TCTM-TD-002

Test Case Title: Verify that reminders can be customized by urgency level.

Test Case Description: As a user, I want to customize the urgency of my reminders for better time management.

Test Suite: Task Management

Test Priority: Medium

Preconditions:

1. User is logged in.

2. User has created a task.

3. User has set a reminder for the task.

Test Data: Selected task with a deadline, a set reminder, and a custom urgency level.

Test Steps:

1. Navigate to the task list.

2. Select a task with a reminder.

3. Click on the "Edit Reminder" button.

4. Change the urgency level of the reminder, e.g. from Low to High.

5. Confirm the urgency level change.

Postconditions:

1. Reminder urgency level is updated for the selected task.

Expected Result: The system updates the urgency of the reminder based on user input.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TCTM-TD-003

Test Case Title: Verify that the system sends notifications for upcoming deadlines.

Test Case Description: As a user, I want to receive notifications for upcoming deadlines so that I can plan my work better.

Test Suite: Task Management

Test Priority: High

Preconditions:

1. User is logged in.
2. User has created a task with a deadline.
3. User has set a reminder for the task.
4. The deadline for the task is within the reminder time frame.

Test Data: Selected task with a deadline within the reminder time frame.

Test Steps:

1. Navigate to the task list.
2. Select a task with an upcoming deadline.

Expected Result: The system sends a notification to the user before the deadline.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TCTM-TD-004

Test Case Title: Verify that users can choose the delivery method for reminders.

Test Case Description: As a user, I want to choose my preferred method of receiving reminders for better accessibility.

Test Suite: Task Management

Test Priority: Medium

Preconditions:

1. User is logged in.
2. User has created a task with a deadline.
3. User has set a reminder for the task.

Test Data: Selected task with a deadline, a set reminder, and a preference for the delivery method.

Test Steps:

1. Navigate to the task list.
2. Select a task with a reminder.
3. Click on the "Edit Reminder" button.
4. Change the reminder delivery method, e.g. from Email to In-App.
5. Confirm the delivery method change.

Postconditions:

1. Reminder delivery method is saved for the selected task.

Expected Result: The system updates the delivery method based on user input.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TCTM-TD-005

Test Case Title: Verify that reminder settings can be adjusted or disabled.

Test Case Description: As a user, I want to maintain control over my reminders by adjusting or disabling them based on my needs.

Test Suite: Task Management

Test Priority: Medium

Preconditions:

1. User is logged in.
2. User has created a task with a deadline.
3. User has set a reminder for the task.

Test Data: Selected task with a deadline, a set reminder, and a need for adjustment or disabling.

Test Steps:

1. Navigate to the task list.
2. Select a task with a reminder.
3. Click on the "Edit Reminder" button.
4. Adjust or disable the reminder, e.g. removing the reminder or selecting the "Disabled" option.
5. Confirm the adjustment or disabling.

Postconditions:

1. Reminder is adjusted or disabled for the selected task.

Expected Result: The system saves the adjustment or disables the reminder based on user input.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 24

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Prioritization

Feature description -

Users can prioritize tasks based on their urgency and importance, allowing them to focus on high-priority tasks and manage their workload effectively. Tasks can be categorized into different priority levels, such as high, medium, or low, based on their importance to the project or team.

Userstory title -

As a user, I want to prioritize tasks so that I can manage my workload effectively.

Acceptance criteria -

['Users can set priority levels for tasks (High, Medium, Low).', 'Tasks can be sorted based on priority levels.', 'Users can change the priority level of a task.', 'The system visually distinguishes tasks based on priority.', 'Tasks without a specified priority default to Medium.']

Generated Test Cases:

Test Case ID: TMT-TC-001

Test Case Title: Verify that users can set priority levels for tasks

Test Case Description: This test case aims to verify that users can assign one of the three priority levels (High, Medium, Low) to a task.

Test Suite: Task Creation

Test Priority: High

Preconditions:

- User is logged in
- User is on the task creation page

Test Data: New task with a title, description, and due date

Test Steps:

1. Enter a title for the new task
2. Add a description for the new task
3. Set a due date for the new task
4. Select a priority level (High, Medium, Low) for the new task
5. Click the "Create Task" button

Postconditions:

- New task is created and saved in the system
- Selected priority level is displayed on the new task

Expected Result: The system allows users to assign a priority level to a new task, and the assigned priority level is displayed correctly.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TMT-TC-002

Test Case Title: Verify that tasks can be sorted based on priority levels

Test Case Description: This test case aims to verify that tasks displayed in a list can be sorted in ascending or descending order based on their priority levels.

Test Suite: Task Listing

Test Priority: Medium

Preconditions:

- User is logged in
- User has at least three tasks with different priority levels

Test Data: N/A

Test Steps:

1. Navigate to the task listing page
2. Verify that tasks are displayed in the default order
3. Click the "Sort by Priority" button
4. Verify that tasks are sorted in ascending order based on priority levels (Low, Medium, High)
5. Click the "Sort by Priority" button again
6. Verify that tasks are sorted in descending order based on priority levels (High, Medium, Low)

Postconditions: N/A

Expected Result: The system allows users to sort tasks based on priority levels in ascending or descending order.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TMT-TC-003

Test Case Title: Verify that users can change the priority level of a task

Test Case Description: This test case aims to verify that users can modify the priority level of an existing task.

Test Suite: Task Editing

Test Priority: Medium

Preconditions:

- User is logged in
- User has at least one task with a specific priority level

Test Data: Existing task with a set priority level

Test Steps:

1. Navigate to the task editing page
2. Locate the priority level section for the existing task
3. Change the priority level of the existing task to a different level (e.g., High to Medium or Low)
4. Click the "Save Changes" button

Postconditions:

- Existing task is updated with the new priority level
- New priority level is displayed on the updated task

Expected Result: The system allows users to change the priority level of an existing task, and the updated priority level is displayed correctly.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TMT-TC-004

Test Case Title: Verify that the system visually distinguishes tasks based on priority

Test Case Description: This test case aims to verify that the Task Management Tool visually distinguishes tasks based on their priority levels using different colors or icons.

Test Suite: Task Listing

Test Priority: Low

Preconditions:

- User is logged in
- User has tasks with different priority levels

Test Data: N/A

Test Steps:

1. Navigate to the task listing page
2. Verify that tasks with different priority levels are visually distinguishable from one another

Expected Result: The Task Management Tool visually distinguishes tasks based on their priority levels using different colors or icons, making it easier for users to identify high-priority tasks.

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TMT-TC-005

Test Case Title: Verify that tasks without a specified priority default to Medium

Test Case Description: This test case aims to verify that tasks without a specified priority level default to the Medium priority level.

Test Suite: Task Creation

Test Priority: Low

Preconditions:

- User is logged in
- User is on the task creation page

Test Data: New task with a title, description, and due date but no priority level specified

Test Steps:

1. Enter a title for the new task
2. Add a description for the new task
3. Set a due date for the new task
4. Do not select a priority level for the new task
5. Click the "Create Task" button

Postconditions:

- New task is created and saved in the system
- The task defaults to the Medium priority level

Expected Result: The system defaults tasks without a specified priority level to the Medium priority level.

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 25

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Subtasks

Feature description -

Tasks can be broken down into smaller subtasks, enabling users to divide complex tasks into manageable

components. Subtasks can have their own due dates, assignees, and dependencies, providing granularity and clarity in task execution.

Userstory title -

As a user, I want to create subtasks so that I can break down complex tasks into manageable parts.

Acceptance criteria -

['Users can create subtasks under main tasks.', 'Subtasks can have their own due dates and assignees.', 'The system tracks the progress of each subtask.', 'Subtasks can be marked as complete independently of the main task.', 'Dependencies between subtasks can be set.']

Generated Test Cases:

Test Case 1:

Test Case ID: TMT-ST-001

Test Case Title: Verify that users can create subtasks under main tasks

Test Case Description: This test case verifies if a user is able to create a subtask under a main task.

Test Suite: Subtask Creation

Test Priority: High

Preconditions:

- User is logged in
- User has created a main task

Test Data: Main task with no subtasks

Test Steps:

1. Navigate to the main task
2. Click on the "Add Subtask" button
3. Enter the required subtask details
4. Click on the "Create" button

Postconditions:

- Subtask is created under the main task

- The user is redirected to the updated task page

Expected Result: Subtask is created successfully and displayed under the main task.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TMT-ST-002

Test Case Title: Verify that subtasks can have their own due dates and assignees

Test Case Description: This test case verifies if a user is able to set a unique due date and assignee for a subtask.

Test Suite: Subtask Configuration

Test Priority: High

Preconditions:

- User is logged in
- User has created a main task
- User has created a subtask under the main task

Test Data: Subtask without due date and assignee

Test Steps:

1. Navigate to the subtask
2. Click on the "Edit" button
3. Set a due date for the subtask
4. Assign the subtask to a team member
5. Save the changes

Postconditions:

- The subtask has a unique due date and assignee

Expected Result: Subtask has its own due date and assignee that are different from the main task.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TMT-ST-003

Test Case Title: Verify that the system tracks the progress of each subtask

Test Case Description: This test case verifies if the system automatically tracks the progress of each subtask and updates the main task progress accordingly.

Test Suite: Subtask Progress Tracking

Test Priority: Medium

Preconditions:

- User is logged in
- User has created a main task
- User has created at least two subtasks under the main task
- One subtask is marked as complete

Test Data: Main task with two subtasks, one completed and one in progress

Test Steps:

1. Navigate to the main task
2. Check the progress of the main task

Postconditions:

- The main task progress percentage reflects the completed subtask

Expected Result: The system automatically updates the main task progress based on completed subtasks.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TMT-ST-004

Test Case Title: Verify that subtasks can be marked as complete independently of the main task

Test Case Description: This test case verifies if a user can mark a subtask as complete without affecting the main task status.

Test Suite: Subtask Independence

Test Priority: Medium

Preconditions:

- User is logged in
- User has created a main task
- User has created a subtask under the main task
- Subtask has a due date in the future

Test Data: Main task with a subtask and due date in the future

Test Steps:

1. Navigate to the subtask
2. Click on the "Mark as Complete" button
3. Confirm the action in the pop-up

Postconditions:

- The subtask is marked as complete
- The main task status remains unchanged

Expected Result: Subtask is marked as complete without affecting the main task status.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TMT-ST-005

Test Case Title: Verify that dependencies between subtasks can be set

Test Case Description: This test case verifies if a user can set dependencies between subtasks.

Test Suite: Subtask Dependencies

Test Priority: High

Preconditions:

- User is logged in
- User has created a main task
- User has created at least two subtasks under the main task

Test Data: Main task with two subtasks, no dependencies

Test Steps:

1. Navigate to the first subtask
2. Edit the subtask
3. In the dependencies section, search for the second subtask
4. Set the first subtask as dependent on the second subtask
5. Save the changes

Postconditions:

- The first subtask is dependent on the second subtask

Expected Result: Subtasks have a dependency set up correctly, and the first subtask cannot be marked as complete before the second subtask.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 26

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Dependencies

Feature description -

Users can define dependencies between tasks, specifying the order in which tasks should be completed to ensure smooth workflow and project progression. Task dependencies help in identifying critical paths and managing interdependencies between tasks.

Userstory title -

As a user, I want to set task dependencies so that tasks are completed in the correct sequence.

Acceptance criteria -

['Users can set dependencies between tasks.', 'The system displays dependencies in the task details.', 'Tasks cannot be marked as complete if dependent tasks are not completed.', 'Users can update or remove dependencies as needed.', 'The system provides alerts or warnings if a dependent task is not completed.']

Generated Test Cases:

Test Case ID: TCT-TD-001

Test Case Title: Verify that users can set dependencies between tasks

Test Case Description: This test case verifies the ability of users to define dependencies between different tasks, ensuring smooth workflow and project progression.

Test Suite: Task Dependencies

Test Priority: High

Preconditions:

- User is logged in
- User has access to create tasks
- User has access to edit tasks

Test Data: Two or more tasks with varying deadlines and priorities

Test Steps:

1. Create a new task in the system.
2. Input a descriptive title and necessary details.
3. Set a deadline for the task.
4. Create a second task with a different deadline.
5. Attempt to set a dependency between the two tasks.
6. Save the changes to both tasks.

Postconditions:

- The system displays the dependency in the task details for both tasks

Expected Result: The system allows users to set dependencies between tasks, and the relationship is clearly displayed in the task details.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-TD-002

Test Case Title: Verify that tasks cannot be marked as complete if dependent tasks are not completed

Test Case Description: This test case verifies that users cannot mark a task as complete if any of its dependent tasks have not been completed.

Test Suite: Task Dependencies

Test Priority: High

Preconditions:

- User is logged in
- User has access to create tasks
- User has access to edit tasks
- User has created two tasks with a dependency set up from task A to task B

Test Data: Two tasks with a dependency from task A to task B

Test Steps:

1. Complete task B.
2. Attempt to mark task A as complete.
3. Save the changes.

Postconditions:

- Task A remains marked as incomplete

Expected Result: The system prevents users from marking a task as complete when there are dependent tasks that have not been completed.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Negative

Test Case ID: TCT-TD-003

Test Case Title: Verify that users can update or remove dependencies as needed

Test Case Description: This test case verifies that users have the ability to modify or remove existing dependencies within tasks.

Test Suite: Task Dependencies

Test Priority: Medium

Preconditions:

- User is logged in
- User has access to create tasks
- User has access to edit tasks
- User has created two tasks with a dependency set up from task A to task B

Test Data: Two tasks with a dependency from task A to task B

Test Steps:

1. Access the task details for task A.
2. Edit the existing dependency or remove it.
3. Save the changes.
4. Verify the change by checking task B's dependencies.

Postconditions:

- The dependency has been updated or removed as intended.

Expected Result: The system allows users to update or remove task dependencies according to their needs.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-TD-004

Test Case Title: Verify that the system provides alerts or warnings if a dependent task is not completed

Test Case Description: This test case verifies that the system issues alerts or warnings to users if a dependent task has not been completed as expected.

Test Suite: Task Dependencies

Test Priority: Medium

Preconditions:

- User is logged in

- User has access to create tasks
- User has access to edit tasks
- User has created two tasks with a dependency set up from task A to task B

Test Data: Two tasks with a dependency from task A to task B, where task B has not been completed

Test Steps:

1. Access task A's details.
2. Observe for any alerts or warnings regarding task B's incomplete status.

Postconditions:

- Alerts or warnings are displayed in task A's details

Expected Result: The system notifies users if any dependent tasks have not been completed.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-TD-005

Test Case Title: Verify that the system displays dependencies in the task details

Test Case Description: This test case verifies that the system displays task dependencies in the task details section of each affected task.

Test Suite: Task Dependencies

Test Priority: Low

Preconditions:

- User is logged in
- User has access to create tasks
- User has access to edit tasks
- User has created tasks with dependencies

Test Data: Multiple tasks with various dependencies

Test Steps:

1. Access the details of each task with dependencies.
2. Observe the dependencies section for any linked tasks.

Postconditions:

- Each task's dependencies are visible in its task details

Expected Result: The system clearly shows task dependencies for users in the task details section.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 27

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Comments and Collaboration

Feature description -

Users can collaborate on tasks by leaving comments, attaching files, and engaging in discussions directly within the task interface. This feature facilitates communication and teamwork, allowing team members to provide updates, share feedback, and ask questions related to the task.

Userstory title -

As a user, I want to comment on tasks so that I can collaborate with my team.

Acceptance criteria -

['Users can add comments to tasks.', 'Comments can include text, attachments, and links.', 'Users can reply to and edit their comments.', 'The system notifies users of new comments.', 'Comments are displayed in chronological order.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that users can add comments to tasks

Test Case Description: This test case verifies that a user is able to add a comment to a task, including text, attachments, and links.

Test Suite: Task Comments and Collaboration

Test Priority: High

Preconditions:

- User is logged in
- User has access to a task

Test Data: Text, attachment, link

Test Steps:

1. Navigate to the task interface.
2. Click on the "Add Comment" button.
3. Enter text, attach a file, and include a link in the comment textbox.
4. Click "Post".

Postconditions:

- Comment is displayed on the task interface.
- Attachment and link are visible and accessible.

Expected Result: A new comment is added to the task, including the provided text, attachment, and link.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that users can reply to and edit their comments

Test Case Description: This test case verifies that a user can reply to and edit their own comments.

Test Suite: Task Comments and Collaboration

Test Priority: Medium

Preconditions:

- User is logged in
- User has access to a task with an existing comment

Test Data: Comment text for reply and edit

Test Steps:

1. Navigate to the task interface.
2. Locate an existing comment and click "Reply".
3. Enter a reply and click "Post".
4. Hover over the original comment and click "Edit".
5. Update the comment text and click "Save".

Postconditions:

- Reply is displayed below the original comment
- Edited comment reflects updated text

Expected Result: User can successfully reply to and edit their own comments.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that the system notifies users of new comments

Test Case Description: This test case verifies that a user receives a notification when there is a new comment on a task they are following.

Test Suite: Task Comments and Collaboration

Test Priority: High

Preconditions:

- User is logged in
- User is following a task
- Another user adds a comment to the task

Test Data: N/A

Test Steps:

1. Log in as the user following the task.
2. Wait for another user to add a comment to the task.

Postconditions:

- User receives a notification of the new comment

Expected Result: User receives a notification when there is a new comment on a task they are following.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that comments are displayed in chronological order

Test Case Description: This test case verifies that comments are displayed in reverse chronological order, with the most recent comment first.

Test Suite: Task Comments and Collaboration

Test Priority: Medium

Preconditions:

- User is logged in
- User has access to a task with multiple comments

Test Data: N/A

Test Steps:

1. Navigate to the task interface.
2. Scroll through the comments section.

Postconditions:

- Comments are displayed with the most recent comment at the top

Expected Result: Comments are displayed in reverse chronological order.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that users cannot edit or delete other users' comments

Test Case Description: This test case verifies that a user cannot edit or delete another user's comment.

Test Suite: Task Comments and Collaboration

Test Priority: Low

Preconditions:

- User is logged in
- User has access to a task with another user's comment

Test Data: N/A

Test Steps:

1. Navigate to the task interface.
2. Locate another user's comment and hover over it.
3. Check for the absence of "Edit" and "Delete" options.

Postconditions:

- User cannot edit or delete another user's comment

Expected Result: A user cannot edit or delete another user's comment.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Negative

Data - 28

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

File Attachments

Feature description -

Users can attach relevant files, documents, and resources to tasks, ensuring that all necessary information is accessible in one place. File attachments can include documents, spreadsheets, images, links, and other resources needed for task execution.

Userstory title -

As a user, I want to attach files to tasks so that all relevant information is easily accessible.

Acceptance criteria -

['Users can attach multiple files to a task.', 'Attachments can include various file types (documents, images, etc.).', 'Users can view and download attachments.', 'The system limits the size of attachments to prevent data overload.', 'Users can remove attachments if needed.']

Generated Test Cases:

Test Case ID: TCT-FA-001

Test Case Title: Verify that users can attach multiple files to a task

Test Case Description: This test case verifies that users are able to attach multiple files to a single task within the task management tool.

Test Suite: File Attachments

Test Priority: High

Preconditions:

- User is logged in
- User is on the task details page

Test Data: Three different file types (e.g. .docx, .jpg, .pdf)

Test Steps:

1. Navigate to the task details page
2. Click on the "Attach File" button
3. Select the first file from the system's file explorer
4. Click "Open" to attach the file
5. Repeat steps 2-4 for the remaining two files

Postconditions:

- Files are displayed in the attachments section
- Files can be viewed and downloaded

Expected Result: Multiple files of different types are successfully attached to the task.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-FA-002

Test Case Title: Verify that attachments can include various file types

Test Case Description: This test case verifies that users can attach various file types, such as documents, images, and links, to tasks.

Test Suite: File Attachments

Test Priority: Medium

Preconditions:

- User is logged in
- User is on the task details page

Test Data: Files of various types (e.g. .docx, .jpg, .pdf, .url)

Test Steps:

1. Navigate to the task details page
2. Click on the "Attach File" button
3. Select the first file from the system's file explorer
4. Click "Open" to attach the file
5. Repeat steps 2-4 for the remaining files of various types

Postconditions:

- Files are displayed in the attachments section
- Files can be viewed and downloaded

Expected Result: Various file types are successfully attached to the task, and can be viewed and downloaded.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-FA-003

Test Case Title: Verify that users can view and download attachments

Test Case Description: This test case verifies that users can view and download attachments added to tasks.

Test Suite: File Attachments

Test Priority: High

Preconditions:

- User is logged in
- User is on the task details page
- A file is attached to the task

Test Data: No test data needed

Test Steps:

1. Navigate to the task details page
2. Locate the attached file within the attachments section
3. Click on the file name or icon to begin the download process

Postconditions:

- File is downloaded to the user's system
- File can be opened and viewed with a compatible application

Expected Result: Attached files can be successfully downloaded and viewed using compatible applications.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-FA-004

Test Case Title: Verify that the system limits the size of attachments

Test Case Description: This test case verifies that the system imposes a file size limit to prevent data overload.

Test Suite: File Attachments

Test Priority: Medium

Preconditions:

- User is logged in

Test Data: File exceeding the size limit

Test Steps:

1. Navigate to the task details page
2. Click on the "Attach File" button
3. Select the file exceeding the size limit
4. Click "Open" to attempt attachment

Postconditions:

- File is not attached

Expected Result: The system blocks attachment of files exceeding the defined size limit.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Negative

Test Case ID: TCT-FA-005

Test Case Title: Verify that users can remove attachments if needed

Test Case Description: This test case verifies that users are able to remove attachments from tasks.

Test Suite: File Attachments

Test Priority: High

Preconditions:

- User is logged in

- A file is attached to the task

Test Data: No test data needed

Test Steps:

1. Navigate to the task details page
2. Locate the attached file within the attachments section
3. Click on the "Remove" or "Delete" button next to the file
4. Confirm file deletion if prompted

Postconditions:

- File is removed from the attachments list
- File is no longer accessible within the task details page

Expected Result: Users are able to successfully remove attachments from tasks.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 29

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Recurring Tasks

Feature description -

Users can set tasks to repeat at specified intervals, automating the creation of recurring tasks such as weekly meetings, monthly reports, or quarterly reviews. Recurring tasks save time and effort by eliminating the need to create them manually for each occurrence.

Userstory title -

As a user, I want to set recurring tasks so that regular activities are automated.

Acceptance criteria -

['Users can set tasks to recur at specified intervals (daily, weekly, monthly).', 'The system generates instances of recurring tasks automatically.', 'Users can modify the recurrence pattern or cancel the recurrence.', 'The system notifies users of upcoming instances of recurring tasks.', 'Users can view a list of all instances of a recurring task.']

Generated Test Cases:

Test Case ID: TCTM-RT-001

Test Case Title: Verify that users can set tasks to recur at daily intervals

Test Case Description: As a user, I want to create a recurring task that is generated daily, so that I don't have to manually create it every day.

Test Suite: Recurring Tasks

Test Priority: High

Preconditions:

- User is logged in
- User has access to the task creation feature
- User has the necessary permissions to create tasks

Test Data:

- Task name: "Daily stand-up meeting"
- Recurrence pattern: Daily
- Start date: Today

- End date: None

Test Steps:

1. Log in as a user
2. Navigate to the task creation page
3. Enter the task name, "Daily stand-up meeting"
4. Set the recurrence pattern to "Daily"
5. Set the start date to today
6. Leave the end date blank
7. Save the task

Postconditions:

- Task is created with recurrence pattern set to daily
- Task is scheduled to generate every day at the same time
- User can view the list of recurring tasks

Expected Result:

- Task is successfully created with the daily recurrence pattern
- User can see the task listed under "Recurring Tasks"
- The system generates a new instance of the task every day at the same time

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCTM-RT-002

Test Case Title: Verify that users can set tasks to recur at weekly intervals

Test Case Description: As a user, I want to create a recurring task that is generated weekly, so that I only have to create it once.

Test Suite: Recurring Tasks

Test Priority: High

Preconditions:

- User is logged in
- User has access to the task creation feature
- User has the necessary permissions to create tasks

Test Data:

- Task name: "Weekly status report"
- Recurrence pattern: Weekly
- Start date: Today
- End date: None

Test Steps:

1. Log in as a user
2. Navigate to the task creation page
3. Enter the task name, "Weekly status report"
4. Set the recurrence pattern to "Weekly"
5. Set the start date to today
6. Leave the end date blank
7. Save the task

Postconditions:

- Task is created with recurrence pattern set to weekly
- Task is scheduled to generate every week on the same day and time
- User can view the list of recurring tasks

Expected Result:

- Task is successfully created with the weekly recurrence pattern
- User can see the task listed under "Recurring Tasks"
- The system generates a new instance of the task every week on the same day and time

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCTM-RT-003

Test Case Title: Verify that users can set tasks to recur at monthly intervals

Test Case Description: As a user, I want to create a recurring task that is generated monthly, so that I don't have to manually create it every month.

Test Suite: Recurring Tasks

Test Priority: High

Preconditions:

- User is logged in
- User has access to the task creation feature
- User has the necessary permissions to create tasks

Test Data:

- Task name: "Monthly budget review"
- Recurrence pattern: Monthly
- Start date: Today
- End date: None

Test Steps:

1. Log in as a user
2. Navigate to the task creation page
3. Enter the task name, "Monthly budget review"
4. Set the recurrence pattern to "Monthly"
5. Set the start date to today
6. Leave the end date blank
7. Save the task

Postconditions:

- Task is created with recurrence pattern set to monthly
- Task is scheduled to generate on the same day of every month
- User can view the list of recurring tasks

Expected Result:

- Task is successfully created with the monthly recurrence pattern
- User can see the task listed under "Recurring Tasks"
- The system generates a new instance of the task every month on the same day

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCTM-RT-004

Test Case Title: Verify that users can modify the recurrence pattern

Test Case Description: As a user, I want to be able to change the recurrence pattern of a recurring task, so that I can adjust it to my needs.

Test Suite: Recurring Tasks

Test Priority: Medium

Preconditions:

- User is logged in
- User has access to the task management feature
- User has the necessary permissions to modify tasks

Test Data:

- Recurring task: "Weekly status report"
- New recurrence pattern: Every two weeks

Test Steps:

1. Log in as a user
2. Navigate to the task list
3. Find the recurring task, "Weekly status report"
4. Click on the task to edit it
5. Change the recurrence pattern to "Every two weeks"
6. Save the task

Postconditions:

- Recurring task's recurrence pattern is changed
- User can view the updated recurrence pattern

Expected Result:

- Recurring task's recurrence pattern is successfully changed from weekly to every two weeks
- User can see the updated recurrence pattern listed under "Recurring Tasks"

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCTM-RT-005

Test Case Title: Verify that users can cancel the recurrence of a recurring task

Test Case Description: As a user, I want to be able to cancel the recurrence of a recurring task, so that it doesn't generate any more instances.

Test Suite: Recurring Tasks

Test Priority: Medium

Preconditions:

- User is logged in
- User has access to the task management feature
- User has the necessary permissions to modify tasks

Test Data:

- Recurring task: "Monthly budget review"

Test Steps:

1. Log in as a user
2. Navigate to the task list
3. Find the recurring task, "Monthly budget review"
4. Click on the task to edit it
5. Cancel the recurrence pattern
6. Save the task

Postconditions:

- Recurring task's recurrence pattern is cancelled
- User can view the cancelled recurrence pattern

Expected Result:

- Recurring task's recurrence pattern is successfully cancelled
- User can see the cancelled recurrence pattern listed under "Tasks" (i.e. no longer listed under "Recurring Tasks")

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 30

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Tags and Labels

Feature description -

Tasks can be categorized and labeled using tags, making it easier to organize and filter tasks based on different criteria such as project, priority, or status. Tags provide a flexible way to categorize tasks and customize task views based on users' preferences.

Userstory title -

As a user, I want to use tags and labels to categorize tasks so that I can easily organize and filter them.

Acceptance criteria -

['Users can add tags and labels to tasks.', 'Tags can be customized and edited.', 'Tasks can be filtered and sorted based on tags and labels.', 'The system displays tags and labels prominently in the task list.', 'Users can remove tags and labels if needed.']

Generated Test Cases:

Test Case 1:

Test Case ID: TMT-001

Test Case Title: Verify that users can add tags and labels to tasks

Test Case Description: As a user, I want to add a tag or label to a task so that I can categorize it based on my preferences.

Test Suite: Task Management Suite

Test Priority: High

Preconditions:

* User is logged in

* User has access to create tasks

Test Data: Newly created task without any tags or labels

Test Steps:

1. Create a new task
2. Locate the "Tags" or "Labels" field
3. Input a new tag/label or select an existing one
4. Save the task

Postconditions:

- * The tag/label is visible in the task list

Expected Result:

- * The system successfully adds the tag/label to the specified task, and the changes are saved.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TMT-002

Test Case Title: Verify that tags can be customized and edited

Test Case Description: As a user, I want to modify tags that I have previously added so that I can maintain consistency and update categories.

Test Suite: Task Management Suite

Test Priority: Medium

Preconditions:

- * User is logged in
- * User has previously added tags

Test Data: Existing tag that needs to be modified

Test Steps:

1. Navigate to the tasks with the tag to be modified
2. Locate the "Edit Tags" option
3. Modify the tag as desired
4. Save the changes

Postconditions:

- * The tag is updated throughout the system

Expected Result:

- * The system updates the tag and reflects the changes consistently.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TMT-003

Test Case Title: Verify that tasks can be filtered and sorted based on tags and labels

Test Case Description: As a user, I want to use tags and labels as filters so that I can quickly view related tasks and organize my work.

Test Suite: Task Management Suite

Test Priority: High

Preconditions:

- * User is logged in
- * User has previously added tags and labels to tasks

Test Data: Assorted tasks with different tags and labels

Test Steps:

1. Navigate to the task list
2. Use the filter or search function for the desired tag or label

3. Verify that only tasks with the specified tag/label are displayed

Postconditions:

- * The filter or search function works as expected

Expected Result:

- * The task list displays only tasks that match the selected tag or label, making it easy for users to find and manage their tasks.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TMT-004

Test Case Title: Verify that the system displays tags and labels prominently in the task list

Test Case Description: As a user, I want to easily identify tasks with specific tags or labels so that I can quickly locate and manage them.

Test Suite: Task Management Suite

Test Priority: Medium

Preconditions:

- * User is logged in
- * User has previously added tags and/or labels to tasks

Test Data: Tasks with various tags and labels

Test Steps:

1. Navigate to the task list
2. Verify that tags and labels are clearly visible for each task

Postconditions:

- * Tags and labels are easy to read and differentiate in the task list

Expected Result:

- * The system displays tags and labels prominently, allowing users to quickly identify tasks based on their categories and preferences.

Severity: Minor

Priority: Medium

Type of Testing: Usability Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TMT-005

Test Case Title: Verify that users can remove tags and labels if needed

Test Case Description: As a user, I want to remove tags or labels from tasks if they are no longer relevant or accurate so that I can maintain an organized and up-to-date task list.

Test Suite: Task Management Suite

Test Priority: High

Preconditions:

- * User is logged in
- * User has previously added tags and labels to tasks

Test Data: Task with an unnecessary tag or label

Test Steps:

1. Navigate to the task with the unwanted tag or label
2. Locate the "Remove Tags" or "Remove Labels" option
3. Select the tag or label to be removed
4. Save the changes

Postconditions:

- * The specified tag or label is no longer associated with the task

Expected Result:

* The system successfully removes the tag or label from the task as desired.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Negative

Data - 31

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Status Tracking

Feature description -

Users can track the status of tasks, including their progress, completion percentage, and current stage. Task status updates provide visibility into task lifecycle and project progression, helping users monitor task progress and identify bottlenecks.

Userstory title -

As a user, I want to track the status of tasks so that I can monitor progress and identify bottlenecks.

Acceptance criteria -

['Users can update the status of tasks (e.g., To Do, In Progress, Completed).', 'The system displays task status prominently in the task list.', 'Users can view the progress percentage of tasks.', 'The system provides notifications for status changes.', 'Task status can be filtered and sorted.']

Generated Test Cases:

Test Case 1:

Test Case ID: TCTS001

Test Case Title: Verify that users can update the status of tasks

Test Case Description: Verify that the system allows users to change the status of tasks from "To Do" to "In Progress" and "Completed."

Test Suite: Task Status Updating

Test Priority: High

Preconditions: User must be logged in, task must be assigned to the user

Test Data: A task with a "To Do" status

Test Steps:

1. Log in as a user.
2. Navigate to the assigned task list.
3. Select a task with a "To Do" status.
4. Click on the "Change Status" button.
5. Choose "In Progress" as the new status.
6. Save the changes.

Postconditions: Task status is updated to "In Progress" and displayed in the task list.

Expected Result: The system should successfully update the status of the task and display the updated status in the task list.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TCTS002

Test Case Title: Verify that task status is displayed prominently

Test Case Description: Verify that the system shows the task status clearly in the task list to facilitate easy tracking.

Test Suite: Task Status Display

Test Priority: Medium

Preconditions: User must be logged in, tasks must be present in the task list

Test Data: List of tasks with varying statuses

Test Steps:

1. Log in as a user.
2. Navigate to the task list.
3. Observe the status of each task in the list.

Postconditions: Task status is displayed clearly and conspicuously for each task.

Expected Result: The system should show the task status prominently for each task in the list.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TCTS003

Test Case Title: Verify that users can view the task progress percentage

Test Case Description: Verify that users can see the progress percentage of tasks based on the total number of sub-tasks or milestones.

Test Suite: Task Progress Display

Test Priority: Medium

Preconditions: User must be logged in, tasks with sub-tasks or milestones must be present in the task list

Test Data: Task with multiple sub-tasks or milestones

Test Steps:

1. Log in as a user.
2. Navigate to the task list.
3. Select a task with sub-tasks or milestones.
4. Observe the progress percentage information.

Postconditions: Progress percentage for the task is displayed accurately.

Expected Result: The system should show the progress percentage based on the completed sub-tasks or milestones.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TCTS004

Test Case Title: Verify that the system provides notifications for status changes

Test Case Description: Verify that the system sends out notifications to the task assignee and relevant stakeholders when a task status is updated.

Test Suite: Status Change Notification

Test Priority: High

Preconditions: User must be logged in, task must be assigned to another user (stakeholder)

Test Data: Task status update event

Test Steps:

1. Log in as a user.
2. Navigate to the assigned task.
3. Update the task status.

Postconditions: Notification is received by the task assignee (stakeholder).

Expected Result: The system should send a notification to the task assignee (stakeholder) when a task status is changed.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TCTS005

Test Case Title: Verify that task status can be filtered and sorted

Test Case Description: Verify that users can view tasks with specific statuses or sort tasks based on their statuses in the task list.

Test Suite: Task Status Filter and Sort

Test Priority: High

Preconditions: User must be logged in, tasks must be present in the task list

Test Data: Tasks with different statuses

Test Steps:

1. Log in as a user.
2. Navigate to the task list.
3. Apply the filter to show tasks with a specific status.
4. Sort tasks based on their statuses.

Postconditions: Tasks with the selected status are displayed, and tasks are sorted by their statuses.

Expected Result: The system should allow users to filter tasks based on their statuses and sort tasks based on their sorted statuses.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 32

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Filtering and Sorting

Feature description -

Users can filter and sort tasks based on various criteria such as due date, priority, assignee, or project. Filtering and sorting options help users focus on specific tasks or view tasks from different perspectives, enhancing task management and organization.

Userstory title -

As a user, I want to filter and sort tasks so that I can focus on specific tasks or view tasks from different perspectives.

Acceptance criteria -

['Users can filter tasks based on criteria such as due date, priority, assignee, or project.', 'Tasks can be sorted by various attributes.', 'The system provides multiple filter and sort options in the task list.', 'Users can save and apply custom filter and sort settings.', 'Filters and sorts do not affect the actual data, only the view.']

Generated Test Cases:

Test Case ID: TST-001

Test Case Title: Verify that users can filter tasks based on due date

Test Case Description: Validate that the Task Management Tool allows users to filter tasks according to their due dates.

Test Suite: Task Filtering

Test Priority: High

Preconditions:

- * User has created tasks with various due dates

- * User is in the task list page

Test Data: Tasks with different due dates

Test Steps:

1. Navigate to the task list page
2. Click on the "Filter by Due Date" button
3. Select a specific due date from the filter options
4. Observe the displayed tasks

Postconditions:

- * Only tasks with the selected due date are shown

Expected Result:

- * Users can filter tasks based on due date

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TST-002

Test Case Title: Verify that tasks can be sorted by priority

Test Case Description: Check if the Task Management Tool enables users to sort tasks based on priority.

Test Suite: Task Sorting

Test Priority: Medium

Preconditions:

- * User has created tasks with various priorities

- * User is in the task list page

Test Data: Tasks with different priorities

Test Steps:

1. Navigate to the task list page
2. Click on the "Sort by Priority" button
3. Observe the sorted tasks

Postconditions:

- * Tasks are sorted from highest to lowest priority (or vice versa)

Expected Result:

- * Users can sort tasks based on priority

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TST-003

Test Case Title: Verify that users can save and apply custom filter and sort settings

Test Case Description: Validate that the Task Management Tool allows users to save and apply custom filter and sort settings.

Test Suite: Custom Filtering and Sorting

Test Priority: High

Preconditions:

- * User is in the task list page

* User has applied and customized task filters and sorts

Test Data: Custom filter and sort settings

Test Steps:

1. Navigate to the task list page
2. Apply custom filters and sorts
3. Click on the "Save Custom Settings" button
4. Choose a name for the custom settings
5. Apply the saved custom settings
6. Observe the displayed tasks

Postconditions:

* Tasks are displayed according to the saved custom filter and sort settings

Expected Result:

* Users can save and apply custom filter and sort settings

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TST-004

Test Case Title: Verify that filters and sorts do not affect the actual data

Test Case Description: Make sure that the changes in filters and sorts do not affect the original data in the Task Management Tool.

Test Suite: Task Filtering and Sorting

Test Priority: High

Preconditions:

* User has created tasks with different data points

* User is in the task list page

Test Data: Tasks with different data points

Test Steps:

1. Navigate to the task list page
2. Apply filters and sorts
3. Update a task
4. Apply different filters or sorts
5. Observe the updated task and other tasks

Postconditions:

- * Updated task remains unchanged in other filter or sort options

Expected Result:

- * Filters and sorts only affect the task list view, not the actual task data

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Negative

Test Case ID: TST-005

Test Case Title: Verify that users can clear filter and sort options

Test Case Description: Validate that the Task Management Tool provides users the ability to clear applied filter and sort options.

Test Suite: Task Filtering and Sorting

Test Priority: Medium

Preconditions:

- * User has applied filters and sorts
- * User is in the task list page

Test Data: N/A

Test Steps:

1. Navigate to the task list page
2. Apply filters and sorts
3. Click on the "Clear Filters and Sorts" button
4. Observe the displayed tasks

Postconditions:

* All filters and sorts are reset

Expected Result:

* Users can clear filter and sort options

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 33

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Time Tracking

Feature description -

Users can track the time spent on individual tasks, enabling accurate time management, project billing, and resource allocation. Time tracking features can include timers, timesheets, and reports that capture billable

hours and productivity metrics.

Userstory title -

As a user, I want to track the time spent on tasks so that I can manage my time efficiently.

Acceptance criteria -

['Users can start and stop a timer for each task.', 'The system records and displays total time spent on tasks.', 'Users can manually enter or adjust time entries.', 'Time tracking data is stored and can be exported.', 'The system provides reports on time spent per task or project.']

Generated Test Cases:

Test Case ID: TC_TIMETRACKING_001

Test Case Title: Verify that users can start and stop a timer for each task

Test Case Description: This test case verifies whether the system allows users to start and stop a timer for individual tasks while recording the elapsed time accurately.

Test Suite: Time Tracking

Test Priority: High

Preconditions:

- User is logged in
- A task has been created

Test Data: No test data needed

Test Steps:

1. Navigate to the task list
2. Select a task to track time
3. Click on the "Start Timer" button next to the task
4. Wait for a few minutes
5. Click on the "Stop Timer" button next to the task
6. Verify that the system displays the elapsed time since the timer was started

Postconditions:

- The system records the elapsed time
- The recorded time is associated with the correct task

Expected Result: The system displays the accurate elapsed time between starting and stopping the timer for an individual task.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC_TIMETRACKING_002

Test Case Title: Verify that the system records and displays total time spent on tasks

Test Case Description: This test case checks whether the system accurately records and displays the total time spent on individual tasks, including time tracked with the timer and manually entered time.

Test Suite: Time Tracking

Test Priority: High

Preconditions:

- User is logged in
- A task has been created

Test Data: Task with some time tracked and some manually entered time

Test Steps:

1. Start the timer for the task
2. Perform other tasks for a few minutes
3. Stop the timer for the task
4. Manually add additional time to the task
5. View the task details
6. Check the total time displayed for the task

Postconditions:

- The system displays the correct total time spent on the task

Expected Result: The system accurately displays the total time spent on a task, including time tracked with the timer and manually entered time.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC_TIMETRACKING_003

Test Case Title: Verify that users can manually enter or adjust time entries

Test Case Description: This test case ensures that users can manually enter or adjust time entries for tasks when necessary.

Test Suite: Time Tracking

Test Priority: Medium

Preconditions:

- User is logged in
- A task has been created

Test Data: Task without any time entry

Test Steps:

1. Navigate to the task list
2. Select a task without a time entry
3. Click on the "Add Time" button
4. Enter the desired time spent on the task manually
5. Save the changes
6. View the task details
7. Verify that the manually entered time is displayed correctly

Postconditions:

- The system saves the manually entered time for the task

Expected Result: Users can manually enter or adjust time entries for tasks as needed.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC_TIMETRACKING_004

Test Case Title: Verify that time tracking data is stored and can be exported

Test Case Description: This test case confirms that time tracking data is stored correctly in the system and can be exported in a suitable format.

Test Suite: Time Tracking

Test Priority: Medium

Preconditions:

- User is logged in
- A task has been created and time has been tracked for it

Test Data: Time tracking data for one or more tasks

Test Steps:

1. Track time for several tasks
2. Navigate to the time tracking report
3. Choose the desired time range and tasks
4. Export the report as a CSV or PDF file
5. Open the exported file

Postconditions:

- The system exports the report correctly
- The report contains accurate time tracking data

Expected Result: Time tracking data is stored correctly and can be exported in a suitable format.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing and Non-Functional Testing (Exporting feature)

Test Case Approach: Positive

Test Case ID: TC_TIMETRACKING_005

Test Case Title: Verify that the system provides reports on time spent per task or project

Test Case Description: This test case validates that the system generates accurate reports on time spent per task or project.

Test Suite: Time Tracking

Test Priority: High

Preconditions:

- User is logged in
- Multiple tasks have been created and time has been tracked for them

Test Data: Time tracking data for several tasks within one or more projects

Test Steps:

1. Navigate to the time tracking report
2. Choose the desired time range and tasks/projects
3. Generate the report for time spent per task or project
4. Check the accuracy of the report

Postconditions:

- The system generates an accurate report

Expected Result: The system generates accurate reports on time spent per task or project.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 34

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Templates

Feature description -

Users can create and use templates for common task types or project workflows, streamlining task creation and standardizing processes across projects. Task templates can include predefined task structures, descriptions, assignees, and due dates, saving time and ensuring consistency in task execution.

Userstory title -

As a user, I want to use task templates so that I can streamline task creation and ensure consistency.

Acceptance criteria -

['Users can create and save task templates.', 'Templates can include predefined task structures, descriptions, assignees, and due dates.', 'Users can apply templates to new tasks or projects.', 'Templates can be edited or deleted as needed.', 'The system displays a list of available templates for selection.']

Generated Test Cases:

Test Case 1:

Test Case ID: TCT-001

Test Case Title: Verify that users can create and save task templates

Test Case Description: This test case verifies that users can create a new task template and save it for future use.

Test Suite: Task Template Creation

Test Priority: High

Preconditions: User is logged in, User has access to the task templates feature

Test Data: New task template details (name, predefined task structure, description, assignees, due dates)

Test Steps:

1. Navigate to the task templates feature
2. Click on "Create New Template"
3. Enter the required information for the new template
4. Save the new template

Postconditions: New task template is saved and displayed in the list of available templates

Expected Result: A new task template is created and saved, allowing users to access it later.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TCT-002

Test Case Title: Verify that templates can include predefined task structures, descriptions, assignees, and due dates

Test Case Description: This test case checks if a saved task template can contain all necessary elements, such as predefined task structures, descriptions, assignees, and due dates.

Test Suite: Task Template Content

Test Priority: Medium

Preconditions: User is logged in, User has access to the task templates feature, A saved task template exists

Test Data: Saved task template details (name, predefined task structure, description, assignees, due dates)

Test Steps:

1. Navigate to the task templates feature
2. Select a previously saved task template
3. Verify the predefined task structure, description, assignees, and due dates

Postconditions: Task template elements are correctly displayed

Expected Result: The saved task template includes predefined task structures, descriptions, assignees, and due dates, which are correctly displayed when the template is opened.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TCT-003

Test Case Title: Verify that users can apply templates to new tasks or projects

Test Case Description: This test case checks if users can apply saved task templates to new tasks or projects.

Test Suite: Applying Task Templates

Test Priority: High

Preconditions: User is logged in, User has access to the task templates feature, A saved task template exists

Test Data: Saved task template, New task or project

Test Steps:

1. Navigate to the task or project creation screen
2. Select the desired task template
3. Populate the new task or project with the predefined data from the template

Postconditions: New task or project has predefined information from the selected template

Expected Result: Users can successfully apply a saved task template when creating a new task or project, saving time and ensuring consistency.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TCT-004

Test Case Title: Verify that templates can be edited or deleted as needed

Test Case Description: This test case verifies that users can edit or delete saved task templates if they become outdated or irrelevant.

Test Suite: Editing and Deleting Task Templates

Test Priority: Medium

Preconditions: User is logged in, User has access to the task templates feature, A saved task template exists

Test Data: Saved task template details (name, predefined task structure, description, assignees, due dates)

Test Steps:

1. Navigate to the task templates feature
2. Select a previously saved task template
3. Edit or delete the task template
4. Save changes

Postconditions: Task template is updated, or template is removed from available templates list

Expected Result: Users have the ability to edit or delete saved task templates, maintaining an up-to-date list of relevant templates.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TCT-005

Test Case Title: Verify that the system displays a list of available templates for selection

Test Case Description: This test case ensures that the system presents a clear and accessible list of available task templates for users to choose from during task or project creation.

Test Suite: Displaying Task Templates List

Test Priority: High

Preconditions: User is logged in, User has access to the task templates feature, At least one saved task template exists

Test Data: Saved task templates

Test Steps:

1. Navigate to the task or project creation screen
2. Access the template selection menu (if applicable)
3. Verify that saved task templates are displayed

Postconditions: A list of available task templates is presented for user selection

Expected Result: A list of saved task templates is displayed, allowing users to quickly and easily apply them to new tasks or projects.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 35

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Calendar Integration

Feature description -

Tasks can be synced with users' calendars, providing a unified view of task deadlines and commitments alongside other scheduled events and appointments. Calendar integration helps users manage their time effectively and avoid scheduling conflicts.

Userstory title -

As a user, I want to sync tasks with my calendar so that I can manage my time and avoid scheduling conflicts.

Acceptance criteria -

['Users can sync tasks with external calendars (e.g., Google Calendar, Outlook).', 'The system displays task deadlines in the calendar view.', 'Users can set reminders for tasks via the calendar.', 'Calendar events update automatically when tasks are changed.', 'Users can manage calendar sync settings within the app.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that users can sync tasks with external calendars

Test Case Description: This test case verifies the ability of users to sync their tasks with external calendar applications, such as Google Calendar and Outlook.

Test Suite: Calendar Integration

Test Priority: High

Preconditions:

- User has logged in and is on the task management dashboard
- User has created at least one task with a deadline

Test Data: Task(s) with deadline(s)

Test Steps:

1. Click on the task to open the task details panel
2. Locate and click on the 'Sync to Calendar' button
3. Follow the prompts to authenticate the external calendar application
4. Verify that the task appears in the external calendar application

Postconditions:

- Task has been synced with the external calendar application

Expected Result: Tasks are successfully synced with external calendar applications.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Integration Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that the system displays task deadlines in the calendar view

Test Case Description: This test case verifies that the system presents task deadlines in the calendar view, allowing users to manage their time effectively.

Test Suite: Calendar Integration

Test Priority: Medium

Preconditions:

- User has logged in and is on the calendar view

- User has at least one task with a deadline

Test Data: Task(s) with deadline(s)

Test Steps:

1. Navigate to the calendar view
2. Scroll through the calendar to locate a task deadline
3. Verify that the task deadline appears on the correct date

Postconditions:

- Task deadlines appear correctly on the calendar

Expected Result: Task deadlines are correctly displayed in the calendar view.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, System Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users can set reminders for tasks via the calendar

Test Case Description: This test case verifies that users can create reminders for tasks through the calendar view, ensuring they are notified of upcoming deadlines.

Test Suite: Calendar Integration

Test Priority: Medium

Preconditions:

- User has logged in and is on the calendar view
- User has at least one task without a reminder

Test Data: Task(s) without reminders

Test Steps:

1. Navigate to the calendar view
2. Locate a task without a reminder

3. Click on the task to open the task details panel
4. Set a reminder for the task using the 'Add Reminder' button
5. Verify that the reminder appears in the user's calendar application

Postconditions:

- Reminder has been added to the task

Expected Result: Users can successfully set reminders for tasks via the calendar.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, System Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that calendar events update automatically when tasks are changed

Test Case Description: This test case verifies that calendar events update automatically when tasks are modified within the task management tool.

Test Suite: Calendar Integration

Test Priority: High

Preconditions:

- User has logged in and is on the task management dashboard
- User has at least one task synced with an external calendar application
- Task has a deadline

Test Data: Modified task

Test Steps:

1. Open the task details panel for the task synced with the external calendar application
2. Modify the deadline for the task
3. Save the changes to the task
4. Verify that the deadline in the external calendar application has been updated

Postconditions:

- Calendar event has been updated to match the modified task

Expected Result: Calendar events update automatically when tasks are changed within the task management tool.

Severity: Critical

Priority: High

Type of Testing: Functional Testing, Integration Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that users can manage calendar sync settings within the app

Test Case Description: This test case verifies that users can manage their calendar synchronization settings within the task management tool.

Test Suite: Calendar Integration

Test Priority: Medium

Preconditions:

- User has logged in and is on the task management dashboard
- User has at least one task

Test Data: No test data needed

Test Steps:

1. Navigate to the user settings
2. Locate and click on the 'Calendar Sync Settings' option
3. Modify calendar sync settings (e.g., disable sync, change external calendar application)
4. Save the changes
5. Verify that the new settings have been applied

Postconditions:

- Calendar sync settings have been changed

Expected Result: Users can successfully manage their calendar synchronization settings within the task management tool.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, System Testing

Test Case Approach: Positive

Data - 36

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Duplication

Feature description -

Users can duplicate tasks or task templates, saving time and effort when creating similar tasks for different projects or recurring activities. Task duplication allows users to reuse existing task structures and configurations, maintaining consistency and reducing manual effort.

Userstory title -

As a user, I want to duplicate tasks so that I can save time when creating similar tasks.

Acceptance criteria -

['Users can duplicate tasks or task templates.', 'The duplicated task retains all original details, including descriptions, assignees, and deadlines.', 'Users can modify the duplicated task if needed.', 'The system clearly

indicates that a task has been duplicated.', 'Duplicated tasks are saved and displayed in the task list.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that users can duplicate tasks

Test Case Description: As a user, I want to ensure that I can duplicate an existing task to save time when creating similar tasks.

Test Suite: Task Management

Test Priority: High

Preconditions:

1. User is logged in.
2. User has at least one existing task.

Test Data: An existing task with various details, such as assignees, deadlines, descriptions, and attachments.

Test Steps:

1. Navigate to the task list.
2. Select an existing task.
3. Click the 'Duplicate' button.
4. Confirm the duplication action.

Postconditions:

1. A new task is created with all the details from the original task.
2. The new task is added to the task list.

Expected Result: The user can successfully duplicate an existing task, and the new task retains all original details, including assignees, deadlines, descriptions, and attachments.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that duplicated tasks have unique IDs

Test Case Description: As a user, I want to ensure that the system generates a unique ID for every duplicated task.

Test Suite: Task Management

Test Priority: Medium

Preconditions:

1. User is logged in.
2. User has at least one existing task.

Test Data: An existing task with various details, such as assignees, deadlines, descriptions, and attachments.

Test Steps:

1. Navigate to the task list.
2. Select an existing task.
3. Click the 'Duplicate' button.
4. Confirm the duplication action.

Postconditions: A new task is created with all the details from the original task and a unique ID.

Expected Result: The system generates a unique ID for every duplicated task, ensuring every task can be distinguished from others.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users can modify duplicated tasks

Test Case Description: As a user, I want to ensure that I can modify the details of a duplicated task if needed.

Test Suite: Task Management

Test Priority: High

Preconditions:

1. User is logged in.
2. User has at least one duplicated task.

Test Data: A duplicated task with various details, such as assignees, deadlines, descriptions, and attachments.

Test Steps:

1. Navigate to the task list.
2. Select a duplicated task.
3. Edit task details (e.g., assignee, deadline, description).
4. Save the changes.

Postconditions: The task details are updated according to user inputs.

Expected Result: Users can modify the details of a duplicated task, ensuring flexibility for task customization.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that a notification appears upon task duplication

Test Case Description: As a user, I want to receive a notification when I successfully duplicate a task.

Test Suite: Notification System

Test Priority: Medium

Preconditions:

1. User is logged in.
2. User has at least one existing task.

Test Data: An existing task with various details, such as assignees, deadlines, descriptions, and attachments.

Test Steps:

1. Navigate to the task list.
2. Select an existing task.
3. Click the 'Duplicate' button.
4. Confirm the duplication action.

Postconditions: A notification appears, informing the user of the successful task duplication.

Expected Result: Users receive a notification indicating that they have successfully duplicated a task, ensuring visibility into system actions.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that duplicated tasks do not overwrite existing tasks

Test Case Description: As a user, I want to ensure that duplicating a task will not overwrite or modify existing tasks.

Test Suite: Task Management

Test Priority: High

Preconditions:

1. User is logged in.
2. User has at least one existing task.
3. User has at least one duplicated task.

Test Data: Two existing tasks (original and duplicate) with various details, such as assignees, deadlines, descriptions, and attachments.

Test Steps:

1. Navigate to the task list.

2. Select a duplicated task.
3. Edit task details (e.g., assignee, deadline, description).
4. Save the changes.

Postconditions: The edits only apply to the duplicated task and do not affect the original task.

Expected Result: Duplicating a task does not overwrite or modify existing tasks, ensuring data integrity and consistent user experience.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Negative

Data - 37

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task History

Feature description -

Users can view the history of changes and updates made to tasks, tracking modifications, comments, and other relevant activity. Task history provides a comprehensive audit trail of task-related actions, facilitating accountability, and transparency in task management.

Userstory title -

As a user, I want to view the history of changes to tasks so that I can track updates and modifications.

Acceptance criteria -

['Users can view a detailed history of changes to tasks.', 'The history includes modifications, comments, and other relevant activity.', 'Users can filter the history by type of change or date.', 'The system provides timestamps for all changes.', 'Task history is accessible from the task details page.']

Generated Test Cases:

Test Case ID: TCT-001

Test Case Title: Verify that users can view a detailed history of changes to tasks

Test Case Description: The test case verifies the user's ability to access and view the detailed history of changes and updates made to tasks in the task management tool.

Test Suite: Task History

Test Priority: High

Preconditions:

- User is logged into the task management tool
- A task with multiple changes has been created by the user or another team member

Test Data: An existing task with multiple changes and updates

Test Steps:

1. Navigate to the task details page of the selected task.
2. Locate and click the "Task History" button or link.
3. Review the list of changes displayed in the task history.

Postconditions:

- Task history is displayed and contains modifications, comments, and other relevant activity.

Expected Result: A comprehensive list of changes and updates made to the task, including timestamps, is displayed.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-002

Test Case Title: Verify that users can filter the history by type of change or date

Test Case Description: The test case verifies the user's ability to apply filters on the task history to view specific types of changes or changes made within a certain date range.

Test Suite: Task History

Test Priority: Medium

Preconditions:

- User is logged into the task management tool
- A task with multiple changes has been created by the user or another team member

Test Data: An existing task with multiple changes and updates, a specific type of change, and a date range

Test Steps:

1. Navigate to the task details page of the selected task.
2. Locate and click the "Task History" button or link.
3. Apply a filter based on the type of change or date range.
4. Verify that the filtered changes are displayed correctly.

Postconditions:

- Task history is displayed and contains only the filtered changes.

Expected Result: The user can successfully filter the task history by type of change or date range, showing the relevant changes only.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-003

Test Case Title: Verify that the system provides timestamps for all changes

Test Case Description: The test case verifies that the task management tool includes timestamps for all changes, updates, and modifications in the task history.

Test Suite: Task History

Test Priority: Medium

Preconditions:

- User is logged into the task management tool
- A task with multiple changes has been created by the user or another team member

Test Data: An existing task with multiple changes and updates

Test Steps:

1. Navigate to the task details page of the selected task.
2. Locate and click the "Task History" button or link.
3. Review the list of changes displayed in the task history.
4. Verify that each change includes a timestamp.

Postconditions:

- Task history is displayed, and all changes have corresponding timestamps.

Expected Result: The task history displays timestamps for all changes, updates, and modifications, allowing users to track when each action was taken.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-004

Test Case Title: Verify that task history is accessible from the task details page

Test Case Description: The test case verifies that users can access and view the task history directly from the

task details page.

Test Suite: Task History

Test Priority: Low

Preconditions:

- User is logged into the task management tool
- A task with changes has been created by the user or another team member

Test Data: An existing task with changes

Test Steps:

1. Navigate to the task details page of the selected task.
2. Verify that a link or button for the task history is present.
3. Click the link or button to access the task history.

Postconditions:

- Task history is displayed.

Expected Result: Users can access the task history directly from the task details page, making it easy to review changes and updates.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-005

Test Case Title: Verify that task history remains accessible even if the task is updated

Test Case Description: The test case verifies that users can still access the task history after making additional changes or updates to the task.

Test Suite: Task History

Test Priority: Low

Preconditions:

- User is logged into the task management tool
- A task with changes has been created by the user or another team member
- The user has previously viewed the task history

Test Data: An existing task with changes and updates

Test Steps:

1. Make a new change or update to the task.
2. Navigate back to the task details page of the task.
3. Verify that the link or button for the task history is still present.
4. Click the link or button to access the task history.

Postconditions:

- Task history is displayed and contains the previous changes as well as the most recent update.

Expected Result: Users can continue to access the task history even after making additional changes or updates to the task, ensuring that all actions are tracked and audited.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 38

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Notifications

Feature description -

Users receive notifications for task updates, comments, mentions, and other relevant activity, keeping them informed and engaged with ongoing task progress. Notifications can be delivered via email, mobile push notifications, or in-app alerts, ensuring timely communication and collaboration.

Userstory title -

As a user, I want to receive notifications for task updates so that I can stay informed about ongoing task progress.

Acceptance criteria -

['Users receive notifications for task updates, comments, mentions, and other relevant activities.',
'Notifications can be delivered via email, mobile push notifications, or in-app alerts.', 'Users can customize notification settings based on preferences.', 'Notifications provide detailed information about the activity.',
'The system logs notification history for future reference.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that users receive notifications for task updates

Test Case Description: Validate that when a user creates, completes, or modifies a task, the system sends a notification to the assigned user(s).

Test Suite: Task Notifications

Test Priority: High

Preconditions:

- User has an active account.
- User is assigned to a task.

Test Data: Task details, assigned user's email or device.

Test Steps:

1. Log in as the assigned user.
2. Create, complete, or modify a task.
3. Observe if a notification is received via email, mobile push notification, or in-app alert.

Postconditions: Notification history is logged.

Expected Result: Notification is received, and the user is informed about the task update.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Integration Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that users can customize notification settings

Test Case Description: Ensure that users have the ability to change their notification preferences based on their desired communication channels (email, mobile, in-app).

Test Suite: Task Notifications

Test Priority: Medium

Preconditions: User has an active account.

Test Data: User preferences for email, mobile push notifications, and in-app alerts.

Test Steps:

1. Log in as the user.
2. Access the notification settings.
3. Change the notification preferences.
4. Save the updated settings.

Expected Result: The system acknowledges the changes, and the user receives notifications according to the updated preferences.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing, System Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that notifications provide detailed information

Test Case Description: Validate that notifications include sufficient context about the updated task, such as the task name, description, and category.

Test Suite: Task Notifications

Test Priority: High

Preconditions: A user is assigned to a task, and a task update occurs.

Test Data: Task details, assigned user's email or device.

Test Steps:

1. Log in as the assigned user.
2. Create, complete, or modify a task.
3. Observe the notification content.

Expected Result: Notification includes relevant information about the updated task.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Integration Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that system logs notification history

Test Case Description: Ensure that all notifications, including task updates, are logged and accessible for future reference.

Test Suite: Task Notifications

Test Priority: Medium

Preconditions: User has an active account and receives task notifications.

Test Steps:

1. Log in as a user with notification history.
2. Access the notification history.
3. Observe the list of notifications.

Expected Result: Notification history logs all past notifications, detailing task updates and other activities.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing, System Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that users receive notifications for comments and mentions

Test Case Description: Validate that when a user is mentioned in a task's comment, the user receives a notification regardless of the task status.

Test Suite: Task Notifications

Test Priority: High

Preconditions: User has an active account, is assigned to a task, and another user mentions them in a task comment.

Test Data: Task details, assigned user's email or device, task comment with mention.

Test Steps:

1. Log in as the assigned user.
2. Navigate to a task where you have been mentioned.
3. Observe if a notification is received via email, mobile push notification, or in-app alert.

Expected Result: Notification is received, and the user is informed about the mention in the task comment.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Integration Testing

Test Case Approach: Positive

Data - 39

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Archiving

Feature description -

Completed or inactive tasks can be archived or moved to a separate section, decluttering the task list and maintaining focus on active tasks. Task archiving helps users keep their task lists organized and prioritized, without losing access to important historical data.

Userstory title -

As a user, I want to archive completed tasks so that I can keep my task list focused on active tasks.

Acceptance criteria -

['Users can archive completed or inactive tasks.', 'Archived tasks are moved to a separate section for easy access.', 'Users can search and view archived tasks.', 'The system provides an option to restore archived tasks to the active list.', 'Archived tasks retain all original details and history.']

Generated Test Cases:

Test Case ID: TC-TMT-001

Test Case Title: Verify that users can archive completed tasks

Test Case Description: As a user, I want to archive a task that has been marked as completed, so that I can keep my task list focused on active tasks.

Test Suite: Task Archiving

Test Priority: High

Preconditions:

- * User is logged in
- * A task has been created and marked as completed

Test Data: The specific task that has been marked as completed

Test Steps:

1. Navigate to the task list
2. Select the completed task from the list
3. Click on the "Archive" button
4. Confirm the archive action

Postconditions:

- * The task is removed from the active task list
- * The task appears in the archived tasks section

Expected Result: The task is successfully moved to the archived tasks section, allowing the user to keep their active task list focused on relevant tasks.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-002

Test Case Title: Verify that archived tasks are moved to a separate section

Test Case Description: As a user, I want to ensure that archived tasks are easily accessible, by being moved to

a separate section.

Test Suite: Task Archiving

Test Priority: Medium

Preconditions:

- * User is logged in

- * A task has been archived

Test Data: The archived task

Test Steps:

1. Navigate to the task list
2. Filter the tasks to display the archived tasks
3. Verify that the expected archived tasks are displayed

Postconditions:

- * The archived tasks are displayed in a separate section

Expected Result: The user can easily access archived tasks by filtering the task list, which helps maintain organization and focus on active tasks.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-003

Test Case Title: Verify that users can search and view archived tasks

Test Case Description: As a user, I want to be able to search and view archived tasks, as I might need to access historical data related to completed tasks.

Test Suite: Task Archiving

Test Priority: High

Preconditions:

* User is logged in

* A task has been archived

Test Data: The archived task

Test Steps:

1. Navigate to the archived tasks section
2. Search for a specific archived task using the search feature
3. View the details of the archived task

Postconditions:

* The archived task is displayed with all its original details and history

Expected Result: The user can efficiently search and view archived tasks, ensuring that historical data can be accessed when needed.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-004

Test Case Title: Verify that the system provides an option to restore archived tasks to the active list

Test Case Description: As a user, I want the system to provide an option to restore archived tasks, if I need to make changes or reactivate a task.

Test Suite: Task Archiving

Test Priority: High

Preconditions:

* User is logged in

* A task has been archived

Test Data: The archived task

Test Steps:

1. Navigate to the archived tasks section
2. Select the archived task to be restored
3. Click on the "Restore" button
4. Confirm the restore action

Postconditions:

- * The task is removed from the archived tasks section
- * The task appears in the active task list

Expected Result: The user has the ability to restore archived tasks, enabling them to reactivate tasks or maintain important information if needed.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-005

Test Case Title: Verify that archived tasks retain all original details and history

Test Case Description: As a user, I want to ensure that archived tasks keep all their original details and history, so that historical data remains accurate and accessible.

Test Suite: Task Archiving

Test Priority: High

Preconditions:

- * User is logged in
- * A task has been archived

Test Data: The archived task

Test Steps:

1. Navigate to the archived tasks section
2. View the details of the archived task

3. Verify that all the original details and history (description, deadlines, attachments, categories) are present

Postconditions:

* The archived task retains all its original details and history

Expected Result: Archived tasks maintain all the initial information, providing the user with accurate and comprehensive historical data.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 40

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Guest Access

Feature description -

Users can grant limited access to external collaborators or clients, allowing them to view and collaborate on specific tasks or projects without full platform access. Guest access features provide flexibility in sharing task information and involving external stakeholders in project discussions and updates.

Userstory title -

As a user, I want to grant guest access to external collaborators so that they can view and collaborate on

specific tasks.

Acceptance criteria -

['Users can invite external collaborators as guests with limited access.', 'Guests can view and collaborate on specific tasks or projects.', 'The system provides clear distinctions between guest and full access roles.', 'Users can manage and revoke guest access at any time.', 'Guests cannot access sensitive or restricted areas of the platform.']

Generated Test Cases:

Test Case 1:

Test Case ID: TCTM-GA-001

Test Case Title: Verify that users can invite external collaborators as guests with limited access.

Test Case Description: This test case verifies that a user can invite an external collaborator as a guest, allowing them to view and collaborate on specific tasks or projects.

Test Suite: Guest Access

Test Priority: High

Preconditions:

- User is logged in to the Task Management Tool
- User navigates to the "Guest Access" section

Test Data: External collaborator's email address

Test Steps:

1. Log in to the Task Management Tool.
2. Navigate to the "Guest Access" section.
3. Enter the external collaborator's email address.
4. Set appropriate guest access permissions.
5. Send the invitation.

Postconditions:

- The external collaborator receives the invitation.
- The invited guest appears in the list of guests.

Expected Result: The guest has been successfully invited, and the user can manage and revoke guest access at any time.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TCTM-GA-002

Test Case Title: Verify that guests can view and collaborate on specific tasks or projects.

Test Case Description: This test case confirms that guests can view and collaborate on specific tasks and projects assigned to them by the user.

Test Suite: Guest Access

Test Priority: High

Preconditions:

- User has invited a guest
- Guest has accepted the invitation

Test Data: Assigned tasks/projects for the guest

Test Steps:

1. Log in as the guest.
2. Navigate to the assigned tasks or projects.
3. View and interact with the tasks.

Postconditions:

- The guest can view and collaborate on the assigned tasks/projects.

Expected Result: The guest can successfully view and collaborate on the assigned tasks or projects, and the user can manage and revoke guest access at any time.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TCTM-GA-003

Test Case Title: Verify that clear distinctions between guest and full access roles exist.

Test Case Description: This test case verifies that the system provides clear distinctions between guest and full access roles to prevent unauthorized access.

Test Suite: Guest Access

Test Priority: Medium

Preconditions:

- User is logged in with full access
- Guest is logged in with limited access

Test Data: N/A

Test Steps:

1. Log in as the full access user.
2. Attempt to access restricted areas or features.
3. Log in as the guest with limited access.
4. Attempt to access restricted areas or features.

Postconditions:

- Full access user can access restricted features.
- Guest with limited access cannot.

Expected Result: The system correctly differentiates between guest and full access roles, ensuring that guests cannot access sensitive or restricted areas of the platform.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TCTM-GA-004

Test Case Title: Verify that users can manage and revoke guest access at any time.

Test Case Description: This test case confirms that users can manage and revoke guest access when necessary.

Test Suite: Guest Access

Test Priority: Medium

Preconditions:

- User has invited a guest
- Guest has accepted the invitation

Test Data: N/A

Test Steps:

1. Log in as the user.
2. Navigate to the list of guests.
3. Select the guest to be removed.
4. Revoke guest access.

Postconditions:

- The guest has been removed from the list of guests.
- The guest cannot access assigned tasks/projects anymore.

Expected Result: The user can successfully manage and revoke guest access at any time, maintaining control

over task information sharing.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TCTM-GA-005

Test Case Title: Verify that guests cannot access sensitive or restricted areas of the platform.

Test Case Description: This test case ensures that guests cannot access sensitive or restricted areas of the platform, preserving data security.

Test Suite: Guest Access

Test Priority: High

Preconditions:

- User has invited a guest
- Guest has accepted the invitation

Test Data: N/A

Test Steps:

1. Log in as the guest.
2. Navigate to the restricted areas.

Postconditions:

- Guest cannot access sensitive or restricted areas.

Expected Result: The system correctly prevents guests from accessing sensitive or restricted areas, ensuring data security.

Severity: Critical

Priority: High

Type of Testing: Security Testing

Test Case Approach: Negative

Data - 41

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Import/Export

Feature description -

Users can import tasks from external sources or export tasks to other task management tools or file formats. Import/export capabilities facilitate data migration and integration with other systems, ensuring seamless collaboration and workflow management across platforms.

Userstory title -

As a user, I want to import and export tasks so that I can migrate data between different task management tools.

Acceptance criteria -

['Users can import tasks from external sources in various formats (e.g., CSV, Excel).', 'Users can export tasks to other task management tools or file formats.', 'The system provides clear mapping of fields during import/export.', 'Users can preview imported data before finalizing the process.', 'The system logs import/export activities for future reference.']

Generated Test Cases:

Test Case ID: TC-TMT-001

Test Case Title: Verify that users can import tasks from external sources in CSV format

Test Case Description: This test case verifies that the system allows users to import tasks from an external CSV file and maps the fields correctly during the import process.

Test Suite: Task Import

Test Priority: High

Preconditions:

1. User is logged in to the Task Management Tool.
2. A valid CSV file containing tasks is prepared.

Test Data: CSV file with tasks

Test Steps:

1. Navigate to the 'Import' section of the Task Management Tool.
2. Select 'CSV' as the import format.
3. Click on 'Choose File' and select the prepared CSV file.
4. Click on 'Import' to start the import process.
5. Review the field mapping provided by the system before finalizing the import.
6. Complete the import process.

Postconditions:

1. The imported tasks are visible in the Task Management Tool.
2. The tasks are mapped correctly to the appropriate fields.

Expected Result: The system imports tasks from the CSV file and maps fields accurately, allowing the user to preview the imported data before finalizing the process.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Data-Driven Testing, Import Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-002

Test Case Title: Verify that users can import tasks from external sources in Excel format

Test Case Description: This test case verifies that the system allows users to import tasks from an external Excel file and maps the fields correctly during the import process.

Test Suite: Task Import

Test Priority: High

Preconditions:

1. User is logged in to the Task Management Tool.
2. A valid Excel file containing tasks is prepared.

Test Data: Excel file with tasks

Test Steps:

1. Navigate to the 'Import' section of the Task Management Tool.
2. Select 'Excel' as the import format.
3. Click on 'Choose File' and select the prepared Excel file.
4. Click on 'Import' to start the import process.
5. Review the field mapping provided by the system before finalizing the import.
6. Complete the import process.

Postconditions:

1. The imported tasks are visible in the Task Management Tool.
2. The tasks are mapped correctly to the appropriate fields.

Expected Result: The system imports tasks from the Excel file and maps fields accurately, allowing the user to preview the imported data before finalizing the process.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Data-Driven Testing, Import Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-003

Test Case Title: Verify that users can export tasks to other task management tools or file formats

Test Case Description: This test case verifies that the system allows users to export tasks to another task management tool or a file format.

Test Suite: Task Export

Test Priority: High

Preconditions:

1. User is logged in to the Task Management Tool.
2. There are tasks available in the Task Management Tool.

Test Data: N/A

Test Steps:

1. Navigate to the 'Tasks' section of the Task Management Tool.
2. Select the tasks to be exported.
3. Click on 'Export' and choose the desired file format or task management tool.
4. Save the exported file.

Postconditions:

1. The exported file is saved on the user's device.

Expected Result: The system exports tasks in the selected file format or to the chosen task management tool, maintaining data integrity and mapping during the export process.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Export Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-004

Test Case Title: Verify that users can preview imported data before finalizing the process

Test Case Description: This test case verifies that the system allows users to preview the imported data before

finalizing the import process.

Test Suite: Task Import

Test Priority: Medium

Preconditions:

1. User is logged in to the Task Management Tool.
2. A valid CSV or Excel file containing tasks is prepared.

Test Data: CSV or Excel file with tasks

Test Steps:

1. Navigate to the 'Import' section of the Task Management Tool.
2. Select 'CSV' or 'Excel' as the import format.
3. Click on 'Choose File' and select the prepared CSV or Excel file.
4. Click on 'Import' to start the import process.
5. Review the preview of imported data.
6. Cancel the import process if necessary.

Postconditions:

1. The import process is not finalized.

Expected Result: The system provides a preview of imported data before finalizing the import process, enabling the user to cancel the import if needed.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing, Import Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-005

Test Case Title: Verify that the system logs import/export activities for future reference

Test Case Description: This test case verifies that the system logs import/export activities and that the logs are accessible for future reference.

Test Suite: Import/Export Logging

Test Priority: Medium

Preconditions:

1. User is logged in to the Task Management Tool.
2. The user has performed at least one import or export operation.

Test Data: N/A

Test Steps:

1. Navigate to the 'Settings' or 'Logs' section of the Task Management Tool.
2. Locate the import/export logs.

Test Postconditions:

1. Import/export logs are available for review.

Expected Result: The system logs import/export activities and maintains a record of these logs for future reference, providing details on the user, date, time, file format, and number of tasks imported or exported.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing, Logging Testing

Test Case Approach: Positive

Data - 42

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Prioritization

Feature description -

Users can set and adjust task priorities, helping to prioritize work and manage deadlines effectively. Task prioritization features can include visual indicators, sorting options, and priority-based notifications, providing a clear view of the most critical tasks.

Userstory title -

As a user, I want to set and adjust task priorities so that I can manage my work and deadlines effectively.

Acceptance criteria -

['Users can set and adjust task priorities.', 'Tasks can be sorted and filtered based on priority levels.', 'The system provides visual indicators of task priority.', 'Users receive notifications for high-priority tasks.', 'Priority changes are logged and tracked.']

Generated Test Cases:

Test Case ID: TMT-TP-001

Test Case Title: Verify that users can set task priorities

Test Case Description: This test case verifies that users can set priorities for tasks.

Test Suite: Task Management

Test Priority: High

Preconditions:

- User is logged in
- User has created a task

Test Data: A newly created task

Test Steps:

1. Navigate to the task details page
2. Locate the priority settings section
3. Select a priority level for the task

4. Confirm the priority change by saving the task

Postconditions:

- Task's priority level has been updated
- The updated priority is displayed in the task list and details

Expected Result: The system allows the user to set a priority for the task, with the new priority visible in the task list and details.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TMT-TP-002

Test Case Title: Verify that tasks can be sorted based on priority levels

Test Case Description: This test case verifies that tasks are sorted correctly based on their priority levels.

Test Suite: Task Management

Test Priority: High

Preconditions:

- User is logged in
- Multiple tasks with varying priority levels exist

Test Data: Multiple tasks with varying priorities

Test Steps:

1. Navigate to the task list page
2. Ensure that the sorting option is set to "Priority"
3. Verify that tasks are sorted in the following order (from highest to lowest priority): High, Medium, Low

Postconditions:

- The task list reflects the correct sorting order

Expected Result: The system sorts tasks based on their priority levels, displaying them in the order: High, Medium, Low.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TMT-TP-003

Test Case Title: Verify that the system provides visual indicators of task priority

Test Case Description: This test case checks if the system visually distinguishes tasks based on their priority levels.

Test Suite: Task Management

Test Priority: High

Preconditions:

- User is logged in
- Multiple tasks with varying priority levels exist

Test Data: Multiple tasks with varying priorities

Test Steps:

1. Navigate to the task list page
2. Observe the visual indicators, such as color, symbols, or icons, associated with each task's priority
3. Verify that High-priority tasks have a more prominent visual indicator than Medium-priority tasks, which in turn have a more prominent indicator than Low-priority tasks

Postconditions:

- Visual indicators reflect the task priorities

Expected Result: The system visually distinguishes task priorities with varying levels of prominence (High > Medium > Low) through the use of color, symbols, or icons.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TMT-TP-004

Test Case Title: Verify that users receive notifications for high-priority tasks

Test Case Description: This test case verifies that users receive notifications for high-priority tasks before their deadlines.

Test Suite: Notifications

Test Priority: Medium

Preconditions:

- User is logged in
- A high-priority task with a deadline exists
- User has notifications enabled

Test Data: A high-priority task with a pending deadline

Test Steps:

1. Log in as the user
2. Ensure that the task's deadline is within the specified notification period (e.g., 24 hours)
3. Wait for the notification to be triggered based on the task's deadline and notification settings

Postconditions:

- User receives a notification for the high-priority task

Expected Result: The system sends a notification to the user for the high-priority task before its deadline.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TMT-TP-005

Test Case Title: Verify that priority changes are logged and tracked

Test Case Description: This test case verifies that the system records and displays task priority changes.

Test Suite: Task Management

Test Priority: Medium

Preconditions:

- User is logged in
- A task with an initial priority level exists

Test Data: A task with an initial priority level

Test Steps:

1. Navigate to the task details page
2. Change the task priority from its initial level (e.g., Low) to a different level (e.g., Medium)
3. Save the updated task

Postconditions:

- The priority change is recorded in the task's history
- The updated priority is displayed in the task list and details

Expected Result: The system logs and displays task priority changes, showing the updated priority in the task list and details, and recording the change in the task's history.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 43

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Collaboration

Feature description -

Users can collaborate on tasks with team members, including assigning tasks to others, sharing updates, and communicating through comments or messages. Collaboration features aim to enhance teamwork and coordination on tasks and projects.

Userstory title -

As a user, I want to collaborate on tasks with my team so that we can work together more effectively and stay updated on task progress.

Acceptance criteria -

['Users can assign tasks to team members.', 'Users can share updates and communicate through comments or messages.', 'Task updates and comments are visible to assigned team members.', 'Users receive notifications for task updates and comments.', 'The system provides a clear view of task collaboration history.']

Generated Test Cases:

Test Case ID: TC-TMT-001

Test Case Title: Verify that users can assign tasks to team members

Test Suite: Task Assignment

Test Priority: High

Preconditions:

- User is logged in
- User has created a task

Test Data: No test data needed

Test Steps:

1. Navigate to the task list
2. Select a task that you have created
3. Click on the "Assign" button
4. Enter the name of a team member in the input field
5. Click "Assign"

Postconditions:

- The assigned team member should receive a notification
- The assigned team member should see the task in their task list

Expected Result: The task is successfully assigned to the specified team member

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-002

Test Case Title: Verify that users can share updates and communicate through comments or messages

Test Suite: Task Communication

Test Priority: Medium

Preconditions:

- User is logged in
- User has access to a task with comments or messages enabled

Test Data: "This is a test update message."

Test Steps:

1. Navigate to the task with comments or messages enabled
2. Scroll down to the comment section
3. Enter the test update message in the comment input field
4. Click "Post"

Postconditions:

- The test update message is visible in the comment section

Expected Result: Users can share updates and communicate through comments or messages

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-003

Test Case Title: Verify that task updates and comments are visible to assigned team members

Test Suite: Task Collaboration

Test Priority: High

Preconditions:

- User is logged in
- User is assigned to a task
- The task has updates or comments

Test Data: No test data needed

Test Steps:

1. Navigate to the task list
2. Select the task with updates or comments
3. Verify that the updates and comments are visible

Postconditions:

- Assigned team members see the updates and comments

Expected Result: Task updates and comments are visible to assigned team members

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-004

Test Case Title: Verify that users receive notifications for task updates and comments

Test Suite: Task Notifications

Test Priority: Medium

Preconditions:

- User is logged in
- User is assigned to a task
- Notifications are enabled

Test Data: "This is a test update message."

Test Steps:

1. Navigate to the task list
2. Select the task you are assigned to
3. Post a test update message as another user in the comments section
4. Verify that the user receives a notification for the task update

Postconditions:

- The assigned team member receives a notification

Expected Result: Users receive notifications for task updates and comments

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-005

Test Case Title: Verify that the system provides a clear view of task collaboration history

Test Suite: Task Collaboration History

Test Priority: High

Preconditions:

- User is logged in
- User is assigned to a task with updates and comments

Test Data: No test data needed

Test Steps:

1. Navigate to the task list
2. Select the task with updates and comments
3. Click on "Collaboration History"
4. Verify that the collaboration history is displayed clearly and includes all updates and comments

Postconditions:

- The user can view the collaboration history

Expected Result: The system provides a clear view of task collaboration history

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 44

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Tracking

Feature description -

Users can track the progress of tasks, view task statuses, and monitor deadlines. The task tracking feature aims to provide visibility into task progress and ensure timely completion of tasks.

Userstory title -

As a user, I want to track the progress of tasks so that I can monitor their status and ensure timely completion.

Acceptance criteria -

['Users can view task statuses.', 'Users can track task progress and deadlines.', 'Tasks are updated in real-time.', 'System provides visual indicators for task progress.', '']

Generated Test Cases:

Test Case ID: TCT-001

Test Case Title: Verify that users can view task statuses

Test Case Description: As a user, I want to check the current status of tasks within the Task Management Tool.

Test Suite: Task Tracking

Test Priority: High

Preconditions:

- * User must be logged in
- * A task with an assigned status must exist

Test Data: No test data needed

Test Steps:

1. Log in to the Task Management Tool.
2. Navigate to the task list.
3. Select a task with an assigned status.
4. View the task status.

Postconditions: N/A

Expected Result: The task status is displayed and visible to the user.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-002

Test Case Title: Verify that users can track task progress and deadlines

Test Case Description: As a user, I want to monitor the progress of tasks and their respective deadlines within the Task Management Tool.

Test Suite: Task Tracking

Test Priority: High

Preconditions:

- * User must be logged in
- * A task with a deadline and progress percentage must exist

Test Data: No test data needed

Test Steps:

1. Log in to the Task Management Tool.
2. Navigate to the task list.
3. Select a task with a deadline and progress percentage.
4. View the progress percentage and deadline.
5. Monitor the progress percentage as it changes over time.

Postconditions: N/A

Expected Result: The progress percentage and deadline are displayed and update in real-time.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-003

Test Case Title: Verify that the system provides visual indicators for task progress

Test Case Description: As a user, I want the Task Management Tool to provide visual indicators of task progress.

Test Suite: Task Tracking

Test Priority: Medium

Preconditions:

- * User must be logged in
- * A task with a progress percentage must exist

Test Data: No test data needed

Test Steps:

1. Log in to the Task Management Tool.
2. Navigate to the task list.
3. Select a task with a progress percentage.
4. Observe the visual indicator for task progress.

Postconditions: N/A

Expected Result: A visual indicator, such as a progress bar or color-coding, clearly displays the task progress percentage.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-004

Test Case Title: Verify that tasks are updated in real-time

Test Case Description: As a user, I want the Task Management Tool to update tasks instantly as progress is made.

Test Suite: Task Tracking

Test Priority: Medium

Preconditions:

* User must be logged in

* A task with a progress percentage must exist

Test Data: No test data needed

Test Steps:

1. Log in to the Task Management Tool.
2. Navigate to the task list.

3. Select a task with a progress percentage.
4. Update the task progress percentage.
5. Observe if the task progress percentage updates immediately.

Postconditions: N/A

Expected Result: The task progress percentage updates in real-time after being manually updated.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-005

Test Case Title: Verify that the system handles delayed task updates

Test Case Description: As a user, I want the Task Management Tool to handle delayed task updates gracefully.

Test Suite: Task Tracking

Test Priority: Low

Preconditions:

- * User must be logged in
- * A task with a progress percentage must exist

Test Data: No test data needed

Test Steps:

1. Log in to the Task Management Tool.
2. Navigate to the task list.
3. Select a task with a progress percentage.
4. Simulate a delay in updating the task progress percentage.
5. Observe if the Task Management Tool handles the delayed update without crashing or displaying errors.

Postconditions: N/A

Expected Result: The Task Management Tool handles delayed updates and continues to function as expected.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Negative

Data - 45

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Notifications

Feature description -

Users receive notifications for task-related events, such as new tasks, updates, comments, and upcoming deadlines. The notifications aim to keep users informed and engaged with their tasks.

Userstory title -

As a user, I want to receive notifications for task-related events so that I can stay informed and manage my tasks effectively.

Acceptance criteria -

['Users receive notifications for new tasks.', 'Users receive notifications for task updates and comments.', 'Users receive reminders for upcoming deadlines.', 'Notification settings can be customized by users.', '']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that users receive notifications for new tasks

Test Suite: Task Notifications

Test Priority: High

Preconditions:

- User is logged in
- A new task is created by another user

Test Data: No test data needed

Test Steps:

1. Log in as a user.
2. Ensure that notifications are enabled.
3. Wait for another user to create a new task.
4. Check for a new task notification.

Postconditions:

- User is notified of the new task

Expected Result:

- The user receives a notification for the new task.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that users receive notifications for task updates and comments

Test Suite: Task Notifications

Test Priority: Medium

Preconditions:

- User is logged in
- A task exists with at least one update or comment

Test Data: No test data needed

Test Steps:

1. Log in as a user.
2. Ensure that notifications are enabled.
3. Access a task with an update or comment.
4. Check for a task update or comment notification.

Postconditions:

- User is notified of task update or comment

Expected Result:

- The user receives a notification for the task update or comment.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users receive reminders for upcoming deadlines

Test Suite: Task Notifications

Test Priority: Medium

Preconditions:

- User is logged in
- A task exists with a deadline set
- The deadline is within the reminder period

Test Data: No test data needed

Test Steps:

1. Log in as a user.
2. Ensure that notifications are enabled.
3. Access a task with an upcoming deadline.
4. Check for an upcoming deadline reminder notification.

Postconditions:

- User is notified of the upcoming deadline

Expected Result:

- The user receives a reminder notification for the upcoming deadline.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that notification settings can be customized by users

Test Suite: Task Notifications

Test Priority: Medium

Preconditions:

- User is logged in

Test Data: No test data needed

Test Steps:

1. Log in as a user.
2. Access the user settings.
3. Navigate to the notification settings.
4. Customize the notification preferences.

5. Create or update a task.

Postconditions:

- User receives notifications according to their customized preferences

Expected Result:

- Users can customize their notification settings and receive notifications based on their preferences.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that no notifications are received when notifications are disabled

Test Suite: Task Notifications

Test Priority: Low

Preconditions:

- User is logged in
- Notifications are disabled

Test Data: No test data needed

Test Steps:

1. Log in as a user.
2. Disable notifications.
3. Create, update, or delete tasks, or wait for deadlines.
4. Check for the absence of notifications.

Postconditions:

- User does not receive notifications

Expected Result:

- Users do not receive notifications when they have disabled them.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Negative

Data - 46

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Categorization

Feature description -

Users can categorize tasks into different groups or projects for better organization and management. Task categorization aims to help users organize tasks by type, priority, or project.

Userstory title -

As a user, I want to categorize tasks into different groups or projects so that I can organize and manage my tasks more effectively.

Acceptance criteria -

['Users can create and assign tasks to different categories or projects.', 'Users can view tasks grouped by categories or projects.', 'Tasks can be moved between categories or projects.', 'Category or project information is accurately reflected in task views.', '']

Generated Test Cases:

Test Case ID: TCT-001

Test Case Title: Verify that users can create and assign tasks to different categories or projects

Test Case Description: As a user, I want to create a new task and assign it to a specific category or project so that I can organize my tasks effectively.

Test Suite: Task Creation and Management

Test Priority: High

Preconditions:

- User is logged in
- User has access to create tasks
- Categories or projects have been created

Test Data:

- Task name
- Task description
- Task due date
- Category or project name

Test Steps:

1. Log in as a user.
2. Navigate to the task creation page.
3. Enter a task name, description, and due date.
4. Select a category or project from the dropdown list.
5. Save the task.

Postconditions:

- The task is saved with the assigned category or project.

- The task is displayed under the selected category or project in the task list.

Expected Result:

- The newly created task should be assigned to the selected category or project and displayed accordingly.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Data-Driven Testing

Test Case Approach: Positive

Test Case ID: TCT-002

Test Case Title: Verify that users can view tasks grouped by categories or projects

Test Case Description: As a user, I want to view my tasks organized by categories or projects so that I can easily manage and track my tasks.

Test Suite: Task Listing and Filtering

Test Priority: High

Preconditions:

- User is logged in
- Tasks have been created and assigned to categories or projects

Test Data: N/A

Test Steps:

1. Log in as a user.

2. Navigate to the task list page.
3. Select a category or project from the filter dropdown menu.

Postconditions:

- The task list is updated to display only tasks from the selected category or project.

Expected Result:

- The task list should display tasks grouped by the selected category or project.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: TCT-003

Test Case Title: Verify that tasks can be moved between categories or projects

Test Case Description: As a user, I want to move tasks between categories or projects to reorganize my tasks as needed.

Test Suite: Task Reorganization

Test Priority: High

Preconditions:

- User is logged in
- Tasks have been created and assigned to categories or projects

Test Data:

- Task name
- Source category or project name
- Target category or project name

Test Steps:

1. Log in as a user.
2. Navigate to the task list page.
3. Select a task from the source category or project.
4. Use the "Move Task" option or drag-and-drop functionality to move the task to the target category or project.
5. Save the change.

Postconditions:

- The task is removed from the source category or project.
- The task is added to the target category or project.

Expected Result:

- The task should be successfully moved between categories or projects.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: TCT-004

Test Case Title: Verify that category or project information is accurately reflected in task views

Test Case Description: As a user, I want to ensure that category or project information is displayed correctly in task previews, details, and lists to maintain consistency and organization.

Test Suite: Task Display and Information

Test Priority: Medium

Preconditions:

- User is logged in
- Tasks have been created and assigned to categories or projects

Test Data: N/A

Test Steps:

1. Log in as a user.
2. Navigate to the task list page.
3. Select a task assigned to a category or project.

Expected Result:

- The task preview, details, and list should accurately display the assigned category or project information.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: TCT-005

Test Case Title: Verify that tasks cannot be assigned to non-existent categories or projects

Test Case Description: As a user, I want to ensure that I cannot assign tasks to non-existent categories or projects to prevent errors and maintain organization.

Test Suite: Task Assignment and Validation

Test Priority: Medium

Preconditions:

- User is logged in

Test Data:

- Task name
- Task description
- Task due date
- Non-existent category or project name

Test Steps:

1. Log in as a user.
2. Navigate to the task creation page.
3. Enter a task name, description, and due date.
4. Select a non-existent category or project from the dropdown list.
5. Save the task.

Postconditions:

- The task is not saved with the non-existent category or project.
- An error message is displayed indicating that the category or project does not exist.

Expected Result:

- The system should prevent the user from assigning tasks to non-existent categories or projects and display an appropriate error message.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Negative Testing, Regression Testing

Test Case Approach: Negative

Data - 47

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Search

Feature description -

Users can search for tasks based on various criteria, such as task name, description, category, priority, or due date. The task search feature aims to help users quickly find specific tasks within their project or task list.

Userstory title -

As a user, I want to search for tasks based on various criteria so that I can quickly find specific tasks within my project or task list.

Acceptance criteria -

['Users can search for tasks by name.', 'Users can search for tasks by description.', 'Users can filter search results by category, priority, or due date.', 'Search results are accurate and relevant.', '']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that users can search for tasks by name

Test Case Description: Validate that the task management tool allows users to input a search query in the task search bar and find tasks based on their names.

Test Suite: Search Functionality

Test Priority: High

Preconditions:

- User is logged in
- User has created tasks with different names

Test Data: Various task names

Test Steps:

1. Log in as a user
2. Navigate to the task list or project
3. Enter a task name in the search bar
4. Click the 'Search' button
5. Check the search results

Postconditions:

- User remains logged in
- Search results are displayed

Expected Result: The search results should include only tasks with names that match the entered query.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that users can search for tasks by description

Test Case Description: Validate that the task management tool allows users to input a search query in the task search bar and find tasks based on their descriptions.

Test Suite: Search Functionality

Test Priority: High

Preconditions:

- User is logged in
- User has created tasks with different descriptions

Test Data: Various task descriptions

Test Steps:

1. Log in as a user
2. Navigate to the task list or project
3. Enter a task description in the search bar
4. Click the 'Search' button
5. Check the search results

Postconditions:

- User remains logged in
- Search results are displayed

Expected Result: The search results should include only tasks with descriptions that match the entered query.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users can filter search results by category, priority, or due date

Test Case Description: Validate that the task management tool allows users to apply filters to search results using category, priority, or due date options.

Test Suite: Search Functionality

Test Priority: High

Preconditions:

- User is logged in
- User has created tasks with various categories, priorities, and due dates

Test Data: Tasks with different categories, priorities, and due dates

Test Steps:

1. Log in as a user
2. Navigate to the task list or project
3. Perform a search (using any criteria, e.g., task name or description)
4. Apply filters using category, priority, or due date options
5. Check the filtered search results

Postconditions:

- User remains logged in
- Filtered search results are displayed

Expected Result: The filtered search results should include only tasks that meet the selected filter criteria (category, priority, or due date).

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that search results are accurate and relevant

Test Case Description: Validate that the task management tool displays accurate and relevant search results based on user queries and filter selections.

Test Suite: Search Functionality

Test Priority: High

Preconditions:

- User is logged in
- User has created tasks with various attributes (names, descriptions, categories, priorities, and due dates)

Test Data: Tasks with various attributes

Test Steps:

1. Log in as a user
2. Navigate to the task list or project
3. Perform a search using different criteria and filter selections
4. Check the search results

Postconditions:

- User remains logged in
- Search results are displayed

Expected Result: Search results should match user queries and filter selections, providing accurate and relevant results.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that users can search for tasks in an empty task list

Test Case Description: Validate that the task management tool handles empty task lists gracefully when users perform a search.

Test Suite: Search Functionality

Test Priority: Medium

Preconditions:

- User is logged in
- User has no tasks in their task list

Test Data: N/A

Test Steps:

1. Log in as a user
2. Ensure the task list is empty
3. Perform a search
4. Check the search results

Postconditions:

- User remains logged in
- A message is displayed indicating no tasks match the query

Expected Result: No tasks match the query message is displayed, indicating that the search function handles empty task lists correctly.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Negative

Data - 48

Project name -

Task Management Tool

Project description -

Build a task management tool that allows users to create, assign, prioritize, and track tasks and projects. Users should be able to set deadlines, add descriptions and attachments to tasks, organize tasks into categories or projects, collaborate with team members, and receive notifications and reminders for upcoming deadlines.

Feature name -

Task Collaboration

Feature description -

Users can collaborate on tasks by adding comments, attaching files, and sharing tasks with other users. Task collaboration aims to enhance team communication and productivity.

Userstory title -

As a user, I want to collaborate on tasks by adding comments, attaching files, and sharing tasks with other users so that I can work more effectively with my team.

Acceptance criteria -

['Users can add comments to tasks.', 'Users can attach files to tasks.', 'Users can share tasks with other users.', 'Task collaboration features work seamlessly and enhance team communication.', '']

Generated Test Cases:

Test Case Creation Template for Task Collaboration Feature:

Test Case ID: TC-TMT-CC-001

Test Case Title: Verify that users can add comments to tasks

Test Case Description: This test case will verify if a user can add comments to a task within the Task Management Tool.

Test Suite: Task Management - Collaboration

Test Priority: High

Preconditions:

1. User is logged in
2. A task has been created
3. User has appropriate permissions

Test Data: Task ID, User login, Comment text

Test Steps:

1. Navigate to the task details page
2. Click on the 'Add Comment' button
3. Enter the comment text in the input field
4. Click on the 'Post Comment' button

Postconditions:

1. Comment is displayed on the task details page
2. Comment is visible in the task's comment history

Expected Result: User should be able to add a comment to a task, and the comment should be visible in the task details page and comment history.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-CC-002

Test Case Title: Verify that users can attach files to tasks

Test Case Description: This test case will verify if users are able to attach files to tasks within the Task Management Tool.

Test Suite: Task Management - Collaboration

Test Priority: Medium

Preconditions:

1. User is logged in

2. A task has been created

3. User has appropriate permissions

Test Data: Task ID, User login, File to be attached

Test Steps:

1. Navigate to the task details page

2. Click on the 'Attach File' button

3. Choose the file to be attached

4. Click on the 'Upload' button

Postconditions:

1. Attached file is displayed in the task details page

2. File is accessible for other users with appropriate permissions

Expected Result: User should be able to attach a file to a task, and the file should be visible and accessible in the task details page for users with appropriate permissions.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-CC-003

Test Case Title: Verify that users receive notifications for task comments and attachments

Test Case Description: This test case will verify if users receive notifications for new task comments and attachments.

Test Suite: Task Management - Collaboration

Test Priority: Medium

Preconditions:

1. User A and User B are logged in

2. User A has created a task

3. User B has appropriate permissions

Test Data: Task ID, User A's login, User B's login, Comment text

Test Steps:

1. User A adds a comment or attachment to the task
2. System sends a notification to User B
3. User B checks the notification

Postconditions:

1. User B received the notification
2. Notification contains the relevant information (task ID, user, comment or attachment)

Expected Result: User B should receive a notification when User A adds a comment or attachment, and the notification should contain relevant information.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-CC-004

Test Case Title: Verify that users cannot add comments or attach files to completed tasks

Test Case Description: This test case will verify that users cannot add comments or attach files to completed tasks.

Test Suite: Task Management - Collaboration

Test Priority: Low

Preconditions:

1. User is logged in
2. A task has been created and marked as completed
3. User has appropriate permissions

Test Data: Task ID, User login

Test Steps:

1. Navigate to the task details page
2. Try to add a comment or attachment to the completed task

Expected Result: Users should not be able to add comments or attachments to completed tasks.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Negative

Test Case ID: TC-TMT-CC-005

Test Case Title: Verify that users can share tasks with other users

Test Case Description: This test case will verify if users can share tasks with other users within the Task Management Tool.

Test Suite: Task Management - Collaboration

Test Priority: Medium

Preconditions:

1. User A and User B are logged in
2. User A has created a task
3. User A has appropriate permissions

Test Data: Task ID, User A's login, User B's login

Test Steps:

1. User A navigates to the task details page
2. User A shares the task with User B by entering their email address
3. User B receives the task sharing invitation
4. User B accepts the task sharing invitation

Postconditions:

1. User B can view the task in their task list

2. User B can edit the task if appropriate permissions are granted

Expected Result: User A should be able to share a task with User B by entering their email address, and User B should receive and accept the task sharing invitation, allowing them to view and edit the task.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 49

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Course Creation Tools

Feature description -

Instructors are equipped with intuitive tools to develop comprehensive courses encompassing various multimedia elements like videos, presentations, quizzes, and assignments. These tools empower instructors to structure content effectively, ensuring an engaging and informative learning experience for students.

Userstory title -

As an instructor, I want intuitive course creation tools so that I can easily develop comprehensive courses.

Acceptance criteria -

['Instructors can create courses using videos, presentations, quizzes, and assignments.', 'The tools are

user-friendly and require minimal technical skills.', 'Courses can be previewed before publishing.', 'Instructors can save courses as drafts.', 'Courses can be edited post-publishing.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC001_CourseCreation_Videos

Test Case Title: Verify that instructors can create courses using videos

Test Case Description: This test case verifies if the course creation tools allow instructors to include video content in their courses.

Test Suite: Course Creation Tools

Test Priority: High

Preconditions:

- User is logged in as an instructor
- User has access to course creation tools

Test Data: A sample video file

Test Steps:

1. Navigate to the course creation page
2. Click on 'Add Content'
3. Select 'Video' and upload the sample video file
4. Save the course changes

Postconditions:

- Video has been added to the course
- Course can be previewed with the video content

Expected Result: The system should allow instructors to include video content in their courses without any issues.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC002_CourseCreation_Presentations

Test Case Title: Verify that instructors can create courses using presentations

Test Case Description: This test case verifies if the course creation tools support presentation files in course content.

Test Suite: Course Creation Tools

Test Priority: High

Preconditions:

- User is logged in as an instructor
- User has access to course creation tools

Test Data: A sample presentation file

Test Steps:

1. Navigate to the course creation page
2. Click on 'Add Content'
3. Select 'Presentation' and upload the sample presentation file
4. Save the course changes

Postconditions:

- Presentation has been added to the course
- Course can be previewed with the presentation content

Expected Result: The system should allow instructors to include presentation content in their courses without any issues.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC003_CourseCreation_UserFriendly

Test Case Title: Verify that course creation tools are user-friendly

Test Case Description: This test case ensures that the course creation tools are intuitive and easy to use for instructors with minimal technical skills.

Test Suite: Course Creation Tools

Test Priority: High

Preconditions:

- User is logged in as an instructor
- User has access to course creation tools

Test Data: No test data needed

Test Steps:

1. Navigate to the course creation page
2. Explore the layout, options, and functionalities available
3. Attempt to add, remove, and modify content elements
4. Preview and save the course changes

Postconditions:

- User has successfully added, removed, or modified content elements
- User is able to understand the layout and options

Expected Result: The course creation tools should be user-friendly and accessible for instructors with various levels of technical proficiency.

Severity: Major

Priority: High

Type of Testing: Usability Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC004_CourseCreation_Preview

Test Case Title: Verify that courses can be previewed before publishing

Test Case Description: This test case checks if the course creation tools allow instructors to preview courses before publishing them.

Test Suite: Course Creation Tools

Test Priority: Medium

Preconditions:

- User is logged in as an instructor
- User has created a course with various content elements

Test Data: No test data needed

Test Steps:

1. Navigate to the course creation page
2. Open the course preview mode
3. Verify that all content elements are displayed correctly

Postconditions:

- User has successfully previewed the course
- Course preview corresponds to the actual content

Expected Result: The system should allow instructors to preview their courses and verify the content before publishing.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC005_CourseCreation_EditPostPublish

Test Case Title: Verify that courses can be edited post-publishing

Test Case Description: This test case validates if courses can be edited after being published.

Test Suite: Course Creation Tools

Test Priority: Low

Preconditions:

- User is logged in as an instructor
- User has published a course

Test Data: No test data needed

Test Steps:

1. Navigate to the course management page
2. Select a published course and enter editing mode
3. Modify a content element, for example, changing a video title
4. Save the course changes

Postconditions:

- Course has been successfully edited
- Edited course is available in the course management page

Expected Result: The system should allow instructors to edit courses after they have been published.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 50

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Interactive Lessons

Feature description -

Lessons are thoughtfully designed to foster learner engagement through interactive components such as quizzes, polls, simulations, and discussions. By encouraging active participation and critical thinking, these interactive elements enhance comprehension and retention of course material.

Userstory title -

As a learner, I want interactive lessons so that I can engage with the content more effectively.

Acceptance criteria -

['Lessons include quizzes, polls, simulations, and discussions.', 'Interactive elements are integrated seamlessly into the lesson content.', 'Learners can track their progress through these activities.', 'Feedback is provided after each interactive component.', 'Interactive lessons can be accessed on various devices.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that lessons include quizzes

Test Case Description: As a learner, I want to ensure that lessons have quizzes integrated so that I can check my understanding of the course material.

Test Suite: Lesson Interactivity

Test Priority: High

Preconditions:

- User is logged in

- User has accessed a lesson

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Navigate to a lesson within a course.
3. Scroll through the lesson content.

Expected Result: At least one quiz should be present within the lesson content.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that interactive elements are integrated seamlessly into the lesson content

Test Case Description: As a learner, I want the interactive elements to be integrated smoothly within the lesson so that my learning experience is not disrupted.

Test Suite: Lesson Interactivity

Test Priority: High

Preconditions:

- User is logged in
- User has accessed a lesson

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Navigate to a lesson within a course.
3. Observe the presentation of quizzes, polls, simulations, and discussions.

Expected Result: Interactive elements should be cohesively integrated into the lesson without causing visual or

functional disruptions.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that learners can track their progress through these activities

Test Case Description: As a learner, I want to monitor my progress through interactive activities so that I can gauge my understanding of the course material.

Test Suite: Progress Tracking

Test Priority: Medium

Preconditions:

- User is logged in
- User has accessed a lesson with interactive activities

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Navigate to a lesson with interactive activities.
3. Engage in the interactive activities provided (quizzes, polls, simulations, etc.)

Expected Result: Completion status of the interactive activities should be accurately reflected and updated in the user's progress.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that feedback is provided after each interactive component

Test Case Description: As a learner, I want to receive feedback after engaging with interactive components so that I can learn from my mistakes and improve my understanding.

Test Suite: Learner Feedback

Test Priority: Medium

Preconditions:

- User is logged in
- User has accessed a lesson with interactive activities

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Navigate to a lesson with interactive activities.
3. Engage in the interactive activities provided (quizzes, polls, simulations, etc.)

Expected Result: Feedback should be displayed after each interactive component, indicating correct or incorrect answers, or relevant information about the user's input.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that interactive lessons can be accessed on various devices

Test Case Description: As a learner, I want interactive lessons to be accessible across multiple devices so that I can learn at my convenience.

Test Suite: Device Compatibility

Test Priority: Low

Preconditions:

- User is logged in

Test Data: No test data needed

Test Steps:

1. Log in as a registered user on different devices (mobile phone, tablet, and desktop).
2. Access a lesson with interactive components on each device.

Expected Result: Interactive lessons should be properly displayed and function as intended on all devices.

Severity: Major

Priority: Low

Type of Testing: Compatibility Testing

Test Case Approach: Positive

Data - 51

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Personalized Learning Paths

Feature description -

Learners have the flexibility to tailor their learning journey by selecting courses, modules, and activities aligned with their unique interests, preferences, and learning objectives. This personalized approach enables learners to pursue their educational goals at their own pace and according to their individual learning styles.

Userstory title -

As a learner, I want personalized learning paths so that I can focus on topics that interest me and progress at my own pace.

Acceptance criteria -

['Learners can choose courses and modules based on their interests.', 'Personalized learning paths adjust dynamically based on learner progress.', 'Learners receive recommendations for new courses and modules.', "Progress tracking is personalized to each learner's path.", 'Learners can adjust their learning paths at any time.']

Generated Test Cases:

Test Case ID: TL-001

Test Case Title: Verify that learners can choose courses and modules based on their interests.

Test Suite: Personalized Learning Paths

Test Priority: High

Preconditions:

- User is logged into the online learning platform.
- User has a clear understanding of their interests and learning objectives.

Test Data: List of courses and modules relevant to the learner's interests.

Test Steps:

1. Log into the online learning platform.
2. Navigate to the course catalog.
3. Filter courses and modules by interest categories.
4. Select a course or module that aligns with the learner's interests.
5. Verify that the selected course or module is added to the learner's personalized learning path.

Postconditions:

- Selected course or module is visible in the learner's personalized learning path.

Expected Result:

- Learners can successfully choose courses and modules based on their interests, which are then added to their personalized learning paths.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TL-002

Test Case Title: Verify that personalized learning paths adjust dynamically based on learner progress.

Test Suite: Personalized Learning Paths

Test Priority: High

Preconditions:

- User is logged into the online learning platform.
- User has a personalized learning path with at least one course or module.

Test Data: User progress data for a course or module.

Test Steps:

1. Log into the online learning platform.
2. Navigate to the personalized learning path.
3. Begin a course or module and complete a portion of it.
4. Verify that the personalized learning path updates to reflect the completed portion of the course or module.
5. Verify that upcoming or recommended courses or modules are adjusted based on the completed portion.

Postconditions:

- Personalized learning path reflects the updated progress and adjusts recommendations accordingly.

Expected Result:

- Personalized learning paths dynamically adjust based on learner progress, updating completed portions and adjusting recommendations accordingly.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TL-003

Test Case Title: Verify that learners receive recommendations for new courses and modules.

Test Suite: Personalized Learning Paths

Test Priority: Medium

Preconditions:

- User is logged into the online learning platform.
- User has a personalized learning path with at least one completed course or module.

Test Data: User progress data and course availability.

Test Steps:

1. Log into the online learning platform.
2. Navigate to the personalized learning path.
3. Verify that a recommendations section exists within the personalized learning path.
4. Verify that at least one recommended course or module is displayed.
5. Click on a recommended course or module and ensure it can be added to the learning path.

Postconditions:

- Learners receive recommendations for new courses and modules within their personalized learning paths.

Expected Result:

- Learners receive personalized recommendations for new courses and modules based on their learning progress and interests.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TL-004

Test Case Title: Verify that progress tracking is personalized to each learner's path.

Test Suite: Personalized Learning Paths

Test Priority: Medium

Preconditions:

- User is logged into the online learning platform.
- User has a personalized learning path with at least one course or module.

Test Data: User progress data and course/module structure.

Test Steps:

1. Log into the online learning platform.
2. Navigate to the personalized learning path.
3. Select a course or module within the learning path.
4. Verify that the progress tracking reflects the user's progress within the selected course or module.
5. Verify that progress tracking for the entire learning path is displayed correctly.

Postconditions:

- Progress tracking accurately reflects each learner's unique progress along their personalized learning path.

Expected Result:

- Progress tracking for each learner's path is personalized and accurately reflects their unique progress within each course or module.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TL-005

Test Case Title: Verify that learners can adjust their learning paths at any time.

Test Suite: Personalized Learning Paths

Test Priority: Medium

Preconditions:

- User is logged into the online learning platform.
- User has a personalized learning path with at least one course or module.

Test Data: List of alternative courses and modules.

Test Steps:

1. Log into the online learning platform.
2. Navigate to the personalized learning path.
3. Select a course or module within the learning path.
4. Remove the selected course or module from the learning path.
5. Add an alternative course or module to the learning path.
6. Verify that the learning path now contains the alternative course or module.

Postconditions:

- Learners can successfully adjust their personalized learning paths as needed.

Expected Result:

- Learners can adjust their learning paths at any time by adding and removing courses or modules.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 52

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Progress Tracking

Feature description -

Learners benefit from comprehensive progress tracking features that provide detailed insights into their advancement through courses, modules, and tasks. By monitoring completed activities, grades, and achievements, learners can gauge their performance and identify areas for improvement.

Userstory title -

As a learner, I want detailed progress tracking so that I can monitor my advancement and identify areas for improvement.

Acceptance criteria -

['Progress tracking shows completed activities and grades.', 'Learners can view detailed reports on their progress.', 'Achievements are highlighted in the progress tracker.', 'Learners receive notifications about their progress.', 'Progress tracking can be exported as reports.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that progress tracking shows completed activities and grades

Test Suite: Progress Tracking Module

Test Priority: High

Preconditions:

- User is logged in
- User is enrolled in a course

Test Data: Course with multiple lectures and assignments

Test Steps:

1. Navigate to the course page
2. Click on a lecture or assignment
3. Complete the lecture or assignment
4. View the progress tracking feature
5. Check if the completed activity is displayed
6. Verify that a grade is assigned (if applicable)

Postconditions:

- Completed activity and grade are displayed in progress tracking

Expected Result: Completed activities and grades are accurately displayed in the progress tracking feature for the specific course.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that learners can view detailed reports on their progress

Test Suite: Progress Tracking Module

Test Priority: High

Preconditions:

- User is logged in
- User is enrolled in a course

Test Data: Course with multiple lectures and assignments

Test Steps:

1. Navigate to the course page
2. Access the progress tracking feature

3. Review the detailed report for the course
4. Check the status of each lecture and assignment (completed or incomplete)
5. Verify any associated grades
6. Identify achievements and milestones

Postconditions:

- Learner can access a detailed report of their progress within the course

Expected Result: The progress tracking feature provides learners with a comprehensive report of their status within the course, including completed and incomplete lectures/assignments and grades.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that achievements are highlighted in the progress tracker

Test Suite: Progress Tracking Module

Test Priority: Medium

Preconditions:

- User is logged in
- User is enrolled in a course
- User has completed specific course requirements (e.g., assignments, quizzes)

Test Data: Course with achievements and milestones

Test Steps:

1. Navigate to the course page
2. Access the progress tracking feature
3. Review the detailed report for the course
4. Identify highlighted achievements and milestones

5. Check the progress requirement for each achievement to ensure completion

Postconditions:

- Achievements are visually distinct and properly highlighted in the progress tracking report

Expected Result: Achievements and milestones are appropriately emphasized within the progress tracking report, confirming completion of specific course requirements.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that learners receive notifications about their progress

Test Suite: Notification System

Test Priority: Medium

Preconditions:

- User is logged in
- User is enrolled in a course
- User has a preferred notification method configured (email, push notification, etc.)

Test Steps:

1. Navigate to the course page
2. Complete a lecture or assignment
3. Wait for the configured notification interval
4. Check the preferred notification method for a progress update

Postconditions:

- Learners receive notifications about their progress according to their preferred notification method

Expected Result: Progress notifications are delivered appropriately based on each learner's preferred notification method.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that progress tracking can be exported as reports

Test Suite: Progress Tracking Module

Test Priority: Low

Preconditions:

- User is logged in
- User is enrolled in a course

Test Steps:

1. Navigate to the course page
2. Access the progress tracking feature
3. Click the "Export" button
4. Select a file format (PDF, CSV, etc.)
5. Save the exported report on the local machine

Postconditions:

- Progress tracking report is downloaded in the selected file format

Expected Result: Learners can export their progress tracking report in the desired file format for offline review and archiving.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach

Data - 53

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Assessment and Grading

Feature description -

Instructors have the ability to create diverse assessments such as quizzes, tests, and assignments to evaluate learner comprehension and mastery of course content. Through timely feedback and accurate grading, instructors support learners in their academic development and provide valuable insights for continuous improvement.

Userstory title -

As an instructor, I want to create diverse assessments so that I can evaluate learner comprehension and provide feedback.

Acceptance criteria -

['Instructors can create quizzes, tests, and assignments.', 'Assessments can include various question types (MCQs, short answers, essays, etc.).', 'Automatic grading is available for objective questions.', 'Manual grading is available for subjective questions.', 'Feedback is provided to learners after grading.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC-001-AG

Test Case Title: Verify that instructors can create quizzes.

Test Suite: Assessment and Grading

Test Priority: High

Preconditions:

- Instructor is logged in.
- Instructor is on the course creation page.

Test Data: No test data needed

Test Steps:

1. Log in as an instructor.
2. Navigate to the course creation page.
3. Click on the "Create Assessment" button.
4. Select "Quiz" as the assessment type.
5. Fill out the required fields (title, description, etc.).
6. Save the quiz.

Postconditions:

- Quiz is saved in the course.

Expected Result:

- Instructor can create a new quiz and save it in the course.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-002-AG

Test Case Title: Verify that assessments can include various question types.

Test Suite: Assessment and Grading

Test Priority: High

Preconditions:

- Instructor is logged in.
- Instructor has created a quiz.

Test Data:

- Pre-populated question data (MCQs, short answers, essays, etc.)

Test Steps:

1. Log in as an instructor.
2. Navigate to a quiz the instructor has created.
3. Click on the "Add Question" button.
4. Select "Multiple Choice Question" as the question type.
5. Enter the question and answer options.
6. Repeat steps 3-5 for short answer, essay, and any other question types.
7. Save the quiz.

Postconditions:

- Quiz includes multiple question types.

Expected Result:

- Instructor can add various question types to a quiz.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-003-AG

Test Case Title: Verify that automatic grading is available for objective questions.

Test Suite: Assessment and Grading

Test Priority: Medium

Preconditions:

- Instructor is logged in.
- Instructor has created a quiz with multiple choice and short answer questions.

Test Data: Pre-populated question data (MCQs, short answers, etc.) with correct answers.

Test Steps:

1. Log in as an instructor.
2. Navigate to a quiz the instructor has created.
3. Allow learners to take the quiz and submit their answers.
4. Access the grading section for the quiz.
5. Verify that multiple choice questions are automatically graded.

6. Manually grade the short answer questions.

Postconditions:

- Objective questions are automatically graded.
- Subjective questions are manually graded.

Expected Result:

- Automatic grading is available for objective questions, such as multiple choice.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-004-AG

Test Case Title: Verify that manual grading is available for subjective questions.

Test Suite: Assessment and Grading

Test Priority: Medium

Preconditions:

- Instructor is logged in.
- Instructor has created a quiz with short answer questions.

Test Data: Pre-populated question data (short answers)

Test Steps:

1. Log in as an instructor.
2. Navigate to a quiz the instructor has created.
3. Allow learners to take the quiz and submit their answers.
4. Access the grading section for the quiz.
5. Manually grade the short answer questions.
6. Save the grades.

Postconditions:

- Subjective questions are manually graded.

Expected Result:

- Manual grading is available for subjective questions, such as short answers.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC-005-AG

Test Case Title: Verify that feedback is provided to learners after grading.

Test Suite: Assessment and Grading

Test Priority: Low

Preconditions:

- Instructor is logged in.

- Instructor has graded a quiz.

Test Data: Pre-populated graded question data (short answers)

Test Steps:

1. Log in as an instructor.
2. Navigate to a quiz the instructor has graded.
3. Access the quiz results page for a learner.
4. Verify that feedback (graded answers, comments, etc.) is displayed.

Postconditions:

- Feedback is visible to learners.

Expected Result:

- Feedback is provided to learners after grading.

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 54

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and

topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Discussion Forums

Feature description -

Learners engage in meaningful discussions, ask questions, and share insights with peers and instructors through dedicated discussion forums. These forums foster collaboration, critical thinking, and knowledge sharing, enriching the learning experience through diverse perspectives and active participation.

Userstory title -

As a learner, I want to participate in discussion forums so that I can engage with peers and instructors.

Acceptance criteria -

['Learners can post questions and responses in forums.', 'Forums support threaded discussions.', 'Instructors can moderate discussions.', 'Learners receive notifications for new posts.', 'Forums are accessible on all devices.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that learners can post questions and responses in forums

Test Case Description: This test case verifies that authenticated learners can create new discussion threads and reply to existing ones in the discussion forums.

Test Suite: Discussion Forums

Test Priority: High

Preconditions:

- User is logged in as a learner
- User is on the course's discussion forum page

Test Data: Prepared question or response text

Test Steps:

1. Navigate to the course's discussion forum page
2. Click the "Start a new discussion" button
3. Enter a question title and text in the provided fields
4. Click the "Post" button
5. Verify that the new discussion thread is visible on the forum page
6. Click on the new discussion thread
7. Enter a response text in the provided field
8. Click the "Reply" button
9. Verify that the response is visible under the discussion thread

Postconditions:

- The new discussion thread and response are saved in the forum

Expected Result: New discussion threads and responses are created and displayed correctly.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that forums support threaded discussions

Test Case Description: This test case verifies that forums support threaded discussions, allowing users to reply to specific posts within a discussion thread.

Test Suite: Discussion Forums

Test Priority: High

Preconditions:

- User is logged in as a learner

- User is on the course's discussion forum page
- A discussion thread with existing replies is present

Test Data: Prepared reply text

Test Steps:

1. Navigate to the course's discussion forum page
2. Locate a discussion thread with existing replies
3. Click the "Reply" button below a specific reply
4. Enter a new reply text in the provided field
5. Click the "Reply" button
6. Verify that the new reply is correctly nested under the reply to which it was posted

Postconditions:

- The new reply is saved and nested correctly

Expected Result: Users can reply to specific posts within a discussion thread, creating a nested, threaded discussion.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that instructors can moderate discussions

Test Case Description: This test case verifies that instructors can moderate discussions by editing, deleting, and pinning posts.

Test Suite: Discussion Forums

Test Priority: High

Preconditions:

- User is logged in as an instructor
- User is on the course's discussion forum page

Test Data: Prepared post text for editing and deletion, test pinning action

Test Steps:

1. Navigate to the course's discussion forum page
2. Create a new discussion thread as an instructor
3. Click the "Edit" button on the discussion thread
4. Modify the text of the discussion thread
5. Click the "Save" button
6. Verify that the discussion thread text has been updated
7. Click the "Delete" button on the discussion thread
8. Confirm the deletion action
9. Verify that the discussion thread no longer appears on the forum page
10. Create a new discussion thread as an instructor
11. Click the "Pin" button on the discussion thread
12. Verify that the pinned discussion thread appears at the top of the forum page

Postconditions:

- Instructor-moderated actions are saved and visible

Expected Result: Instructors can edit, delete, and pin discussion threads in the forums.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that learners receive notifications for new posts

Test Case Description: This test case verifies that learners receive notifications for new posts in discussions they are following or have previously engaged with.

Test Suite: Discussion Forums

Test Priority: Medium

Preconditions:

- User is logged in as a learner
- User is following a discussion thread or has previously engaged with it
- A new post is added to the monitored discussion thread

Test Data: Prepared new post text

Test Steps:

1. Navigate to the course's discussion forum page
2. Verify that no new notification is present
3. Add a new post to a discussion thread being followed
- 4

User is logged in as a learner

User is following a discussion thread or has previously engaged with it

A new post is added to the monitored discussion thread

Prepared new post text

3. Navigate to the course's discussion forum page

4. Verify that a new notification is present

Postconditions:

- A new notification is visible for the new post

Expected Result: Learners receive notifications for new posts in discussions they are following or have previously engaged with.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that forums are accessible on all devices

Test Case Description: This test case verifies that discussion forums can be accessed and interacted with on various devices, including desktops, tablets, and mobile phones.

Test Suite: Discussion Forums

Test Priority: Low

Preconditions:

- User is logged in as a learner or instructor
- User has access to a desktop, tablet, or mobile phone

Test Data: Prepared question or response text

Test Steps:

1. Access the online learning platform on a desktop, tablet, and mobile phone
2. Navigate to the course's discussion forum page on each device
3. Verify that the forum layout is responsive and suitable for the device
4. Perform actions to create, reply, edit, and delete posts on each device
5. Verify that the actions are performed correctly on each device

Postconditions:

- Discussion forums are accessible and functional on various devices

Expected Result: Discussion forums are accessible and can be interacted with on desktops, tablets, and mobile phones.

Severity: Minor

Priority: Low

Type of Testing: Compatibility Testing

Test Case Approach: Positive

Data - 55

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Live Classes and Webinars

Feature description -

The platform facilitates live interactive sessions, allowing instructors to conduct real-time classes and webinars through seamless video conferencing tools. These live sessions enable dynamic engagement, Q&A opportunities, and direct interaction between instructors and learners, replicating the immersive experience of traditional classroom settings.

Userstory title -

As a learner, I want to attend live classes and webinars so that I can interact with instructors in real-time.

Acceptance criteria -

['Live classes and webinars are conducted through video conferencing tools.', 'Learners can ask questions in real-time.', 'Sessions are recorded and available for later viewing.', 'Instructors can share screen and resources during sessions.', 'Notifications are sent before sessions start.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC-LCW-001

Test Case Title: Verify that live classes and webinars are conducted through video conferencing tools

Test Case Description: This test case verifies if the online learning platform supports live classes and webinars using video conferencing tools as specified in the acceptance criteria.

Test Suite: Live Classes and Webinars

Test Priority: High

Preconditions:

- User account is created and logged in.
- Instructor has scheduled a live class or webinar.

Test Data: N/A

Test Steps:

1. Log in as a learner.
2. Access the dashboard or course page.
3. Select the scheduled live class or webinar.
4. Verify if the video conferencing tool is available and accessible within the session.

Postconditions: N/A

Expected Result: The live class or webinar should be conducted using a video conferencing tool, allowing learners to participate in real-time.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-LCW-002

Test Case Title: Verify that learners can ask questions in real-time during live classes and webinars

Test Case Description: This test case validates if learners can submit questions during live sessions and if the instructor can view and respond to them.

Test Suite: Live Classes and Webinars

Test Priority: High

Preconditions:

- User account is created and logged in.
- Instructor has scheduled a live class or webinar.
- Learner has joined the live class or webinar.

Test Data: Predefined questions for the instructor to respond to.

Test Steps:

1. Log in as a learner.
2. Join a live class or webinar.
3. Prepare predefined questions for submission.
4. Submit questions during the live session.
5. Verify if the instructor can view and respond to the questions in real-time.

Postconditions: N/A

Expected Result: Learners should be able to ask questions during live classes or webinars, and instructors should be able to view and respond to them in real-time.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-LCW-003

Test Case Title: Verify that sessions are recorded and available for later viewing

Test Case Description: This test case checks if live classes and webinars are being recorded and if they are accessible for later viewing by learners.

Test Suite: Live Classes and Webinars

Test Priority: High

Preconditions:

- User account is created and logged in.
- Instructor has scheduled and conducted a live class or webinar.

Test Data: N/A

Test Steps:

1. Log in as a learner.
2. Access the dashboard or course page.
3. Locate and select the completed live class or webinar.
4. Verify if the recorded session is available for viewing.

Postconditions: N/A

Expected Result: Completed live classes and webinars should be recorded and accessible for later viewing by learners.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-LCW-004

Test Case Title: Verify that instructors can share screen and resources during live classes and webinars

Test Case Description: This test case validates if instructors can share their screen and resources during live sessions, improving the overall learning experience.

Test Suite: Live Classes and Webinars

Test Priority: High

Preconditions:

- User account is created and logged in.
- Instructor has scheduled a live class or webinar.
- Learner has joined the live class or webinar.

Test Data: Predefined screen sharing materials or resources (e.g. presentations, documents).

Test Steps:

1. Log in as an instructor.
2. Schedule and start a live class or webinar.
3. Join the live class or webinar as a learner using a separate account.
4. Share predefined screen sharing materials or resources during the live session.
5. Verify if the shared materials are visible and accessible to the learner.

Postconditions: N/A

Expected Result: Instructors should be able to share their screen and resources during live classes or webinars, allowing learners to access and view them in real-time.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC-LCW-005

Test Case Title: Verify that notifications are sent before live classes and webinars start

Test Case Description: This test case validate if learners receive notifications before the start of a live class or webinar, ensuring timely attendance and participation.

Test Suite: Live Classes and Webinars

Test Priority: High

Preconditions:

- User account is created and logged in.
- Instructor has scheduled a live class or webinar.

Test Data: N/A

Test Steps:

1. Log in as a learner.
2. Subscribe to a course that includes a live class or webinar.
3. Verify if the notification settings for live classes or webinars are enabled.
4. Wait for the scheduled notification before the live class or webinar starts.
5. Verify if the notification is received.

Postconditions: N/A

Expected Result: Learners should receive notifications before live classes or webinars start, as specified in the acceptance criteria.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 56

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources,

discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Mobile Accessibility

Feature description -

Learners enjoy seamless access to course materials and learning activities via mobile devices, ensuring flexibility and convenience in their learning journey. Whether on-the-go or in remote locations, mobile accessibility enables continuous learning opportunities, empowering learners to maximize their educational experience anytime, anywhere.

Userstory title -

As a learner, I want mobile access to the platform so that I can learn on the go.

Acceptance criteria -

['The platform is accessible on smartphones and tablets.', 'Mobile UI is optimized for touch navigation.', 'Course materials are accessible and readable on mobile devices.', 'Notifications are sent to mobile devices.', 'The platform supports offline access to downloaded materials.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC-ML-001

Test Case Title: Verify that the platform is accessible on smartphones

Test Case Description: Validate if the online learning platform can be accessed and navigated on a smartphone without any functional or UI issues.

Test Suite: Mobile Accessibility Suite

Test Priority: High

Preconditions:

- A stable internet connection
- A compatible smartphone (iOS or Android)
- A registered user account

Test Data: No test data needed

Test Steps:

1. Launch the mobile browser on the smartphone.
2. Enter the URL for the online learning platform.
3. Log in using the registered user account.
4. Browse through different course categories and select a course.
5. Play a video lecture and check for smooth streaming.

Postconditions:

- User remains logged in.
- The platform maintains consistent performance.

Expected Result:

- The platform loads and functions correctly on the smartphone.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Compatibility Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-ML-002

Test Case Title: Verify that mobile UI is optimized for touch navigation

Test Case Description: Ensure that the mobile UI supports intuitive touch navigation and gestures without causing performance issues or incorrect behavior.

Test Suite: Mobile UI Test Suite

Test Priority: Medium

Preconditions:

- A stable internet connection
- A compatible smartphone (iOS or Android)

- A registered user account

Test Data: No test data needed

Test Steps:

1. Launch the mobile browser on the smartphone.
2. Enter the URL for the online learning platform and log in.
3. Test touch interactions, such as menu taps, swipes, and zoom gestures.
4. Try scrolling through long pages and check for lag or performance issues.

Postconditions:

- The platform maintains a smooth user experience.

Expected Result:

- The mobile UI supports touch navigation and gestures effectively.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Usability Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-ML-003

Test Case Title: Verify that course materials are accessible and readable on mobile devices

Test Case Description: Confirm that course materials, including text, images, and videos, are displayed properly and are easily accessible on mobile devices.

Test Suite: Course Material Accessibility Suite

Test Priority: High

Preconditions:

- A stable internet connection
- A compatible smartphone (iOS or Android)
- A registered user account

Test Data: Course materials from various subjects and topics

Test Steps:

1. Log in to the platform on the smartphone.
2. Select a course and access its materials.
3. Verify that text is legible, images are clear, and videos are streamable.
4. Test the download feature for offline access.

Postconditions:

- Downloaded materials are saved to the device.
- The platform maintains consistent performance.

Expected Result:

- Course materials are accessible, readable, and downloadable on mobile devices.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Compatibility Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-ML-004

Test Case Title: Verify that notifications are sent to mobile devices

Test Case Description: Check if the platform sends push notifications regarding course updates, new lectures, or other relevant information to the user's mobile device.

Test Suite: Notification Test Suite

Test Priority: Medium

Preconditions:

- A stable internet connection
- A compatible smartphone (iOS or Android) with the platform app installed
- A registered user account

- Notifications enabled on the smartphone

Test Data: No test data needed

Test Steps:

1. Install and log in to the platform app on the smartphone.
2. Subscribe to notifications for a specific course or topic.
3. Wait for notifications to be sent.
4. Verify that the notifications are received on the smartphone.

Postconditions:

- Notifications are enabled and functioning properly.

Expected Result:

- The platform sends notifications to the user's mobile device.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Non-Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC-ML-005

Test Case Title: Verify that the platform supports offline access to downloaded materials

Test Case Description: Ensure that the platform allows users to download materials for offline access and that such materials can be accessed even when the device is not connected to the internet.

Test Suite: Offline Access Suite

Test Priority: High

Preconditions:

- A stable internet connection
- A compatible smartphone (iOS or Android)
- A registered user account

Test Data: Course materials from various subjects and topics

Test Steps:

1. Log in to the platform on the smartphone.
2. Select a course and download its materials for offline access.
3. Disconnect the device from the internet.
4. Access the downloaded materials from the course.

Postconditions:

- Downloaded materials are available offline.

Expected Result:

- The platform supports offline access to downloaded materials.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Non-Functional Testing

Test Case Approach: Positive

Data - 57

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Resource Library

Feature description -

A comprehensive repository of supplementary materials such as e-books, articles, and reference guides enriches the learning experience by providing additional insights and resources. Learners can access diverse content to deepen their understanding, explore related topics, and enhance their knowledge beyond the confines of the core curriculum.

Userstory title -

As a learner, I want access to a resource library so that I can explore additional materials and deepen my understanding.

Acceptance criteria -

['Resource library includes e-books, articles, and reference guides.', 'Learners can search for specific materials.', 'Resources are categorized and tagged for easy navigation.', 'Learners can download resources for offline access.', 'Instructors can recommend resources to learners.']

Generated Test Cases:

Test Case 1:

Test Case ID: TL-001

Test Case Title: Verify that the resource library includes e-books, articles, and reference guides

Test Case Description: Validate that the resource library contains various types of supplementary materials, such as e-books, articles, and reference guides, in order to confirm that learners have access to a comprehensive repository of additional learning resources.

Test Suite: Resource Library

Test Priority: High

Preconditions:

- User is logged into the online learning platform
- User is on the resource library page

Test Data: N/A

Test Steps:

1. Log into the online learning platform.

2. Navigate to the resource library page.

3. Verify that e-books, articles, and reference guides are available in the resource library.

Postconditions: N/A

Expected Result: The resource library contains e-books, articles, and reference guides, providing learners with a variety of supplementary materials.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TL-002

Test Case Title: Verify that learners can search for specific materials

Test Case Description: Check if the resource library has a search function that allows learners to look for specific resources by title, author, or tag, ensuring that learners can efficiently find and access materials relevant to their learning needs.

Test Suite: Resource Library Search Function

Test Priority: High

Preconditions:

- User is logged into the online learning platform
- User is on the resource library page

Test Data: Sample resource titles or authors

Test Steps:

1. Log into the online learning platform.
2. Navigate to the resource library page.
3. Enter a sample resource title or author into the search bar.
4. Observe if the search results display the desired resource.

Postconditions: N/A

Expected Result: The resource library search function returns accurate and relevant results based on user input.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TL-003

Test Case Title: Verify that resources are categorized and tagged for easy navigation

Test Case Description: Validate that resources in the library are categorized and tagged based on subject, topic, and relevance, enabling learners to easily find and navigate through relevant materials.

Test Suite: Resource Library Categorization

Test Priority: Medium

Preconditions:

- User is logged into the online learning platform
- User is on the resource library page

Test Data: N/A

Test Steps:

1. Log into the online learning platform.
2. Navigate to the resource library page.
3. Verify that resources are categorized by subject and topic.
4. Verify that resources have relevant tags.

Postconditions: N/A

Expected Result: Resources in the library are categorized and tagged appropriately for easy navigation.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TL-004

Test Case Title: Verify that learners can download resources for offline access

Test Case Description: Ensure that learners can download resources from the library to view offline, catering to users who may have limited internet access or wish to study without an internet connection.

Test Suite: Resource Library Download Function

Test Priority: Medium

Preconditions:

- User is logged into the online learning platform
- User is on the resource library page
- Resource is available for download

Test Data: Sample resource for download

Test Steps:

1. Log into the online learning platform.
2. Navigate to the resource library page.
3. Select a sample resource that is available for download.
4. Initiate the download process and save the resource to a desired location.
5. Open the downloaded file to ensure it is not corrupted.

Postconditions: N/A

Expected Result: Resources can be successfully downloaded and accessed offline.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TL-005

Test Case Title: Verify that instructors can recommend resources to learners

Test Case Description: Confirm that instructors can recommend specific resources from the library, helping learners access supplementary materials tailored for the course and enhancing the learning experience.

Test Suite: Resource Library Instructor Recommendation

Test Priority: Low

Preconditions:

- Instructor is logged into the online learning platform
- Instructor is on the course page

Test Data: Sample resource for recommendation

Test Steps:

1. Log into the online learning platform as an instructor.
2. Navigate to the course page.
3. Locate a sample resource in the resource library.
4. Recommend the resource on the course page.
5. Log in as a learner and navigate to the recommended resource on the course page.

Postconditions: N/A

Expected Result: Instructors can recommend resources in the library, allowing learners to access supplementary materials tailored for the course.

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 58

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Peer Review

Feature description -

Learners actively engage in peer review processes, offering constructive feedback and insights on each other's work. Through collaborative assessment and review, learners develop critical evaluation skills, enhance their understanding of course concepts, and benefit from peer-to-peer learning exchanges.

Userstory title -

As a learner, I want to participate in peer review so that I can give and receive feedback on assignments.

Acceptance criteria -

['Learners can submit assignments for peer review.', 'Peers can provide feedback on submitted assignments.', 'A structured review process guides the feedback.', 'Learners can view feedback on their work.', 'Reviewers remain anonymous to ensure unbiased feedback.']

Generated Test Cases:

Test Case ID: TEST-PR-001

Test Case Title: Verify that learners can submit assignments for peer review

Test Case Description: This test case verifies that authenticated learners can successfully submit course assignments for peer review.

Test Suite: Peer Review

Test Priority: High

Preconditions:

- Learner is logged into the online learning platform
- Learner is enrolled in a course with assignment submission functionality

Test Data:

- Assignment file or content

Test Steps:

1. Log into the online learning platform as a learner
2. Navigate to the enrolled course with assignment submission functionality
3. Locate the assignment submission area
4. Upload or input the required assignment file or content
5. Click the "Submit for Peer Review" button

Postconditions:

- Assignment is submitted for peer review
- Learner can view the submitted assignment in their profile

Expected Result: Assignment is successfully submitted for peer review, with a confirmation message displayed to the learner.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TEST-PR-002

Test Case Title: Verify that peers can provide feedback on submitted assignments

Test Case Description: This test case validates that authenticated learners can access and provide constructive feedback on peer assignments.

Test Suite: Peer Review

Test Priority: High

Preconditions:

- Learner is logged into the online learning platform
- Learner is enrolled in a course with active peer review assignments
- Another learner has submitted an assignment for peer review

Test Data:

- Pre-written feedback

Test Steps:

1. Log into the online learning platform as a learner
2. Navigate to the enrolled course with active peer review assignments
3. Locate the available peer review assignment
4. Click on the assignment to view details and provide feedback
5. Input the pre-written feedback in the feedback text box
6. Click the "Submit Feedback" button

Postconditions:

- Feedback is successfully submitted and attached to the assignment
- Learner can view the feedback in the assignment details

Expected Result: Peer feedback is successfully submitted and displayed in the assignment details, with a confirmation message shown to the learner.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TEST-PR-003

Test Case Title: Verify that a structured review process guides the feedback

Test Case Description: This test case checks that a defined structure is present to ensure consistent, constructive feedback on peer assignments.

Test Suite: Peer Review

Test Priority: Medium

Preconditions:

- Learner is logged into the online learning platform
- Learner is enrolled in a course with active peer review assignments

Test Data: N/A

Test Steps:

1. Log into the online learning platform as a learner
2. Navigate to the enrolled course with active peer review assignments
3. Locate the available peer review assignment
4. Click on the assignment to view details and provide feedback
5. Observe the required feedback structure, including (but not limited to) categories, rating scales, or open-ended questions

Expected Result: A consistent feedback structure is present for all peer review assignments, guiding learners in providing constructive input.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TEST-PR-004

Test Case Title: Verify that learners can view feedback on their work

Test Case Description: This test case confirms that learners can access and view feedback provided on their submitted assignments.

Test Suite: Peer Review

Test Priority: High

Preconditions:

- Learner is logged into the online learning platform
- Learner is enrolled in a course with completed peer review assignments
- Another learner has provided feedback on the learner's assignment

Test Data: N/A

Test Steps:

1. Log into the online learning platform as a learner
2. Navigate to the enrolled course with completed peer review assignments
3. Locate the submitted assignment with received feedback
4. Click on the assignment to view details
5. Observe the provided feedback within the assignment details

Expected Result: Learners can view feedback provided by peers on their submitted assignments, allowing them to review and consider the input.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TEST-PR-005

Test Case Title: Verify that reviewers remain anonymous to ensure unbiased feedback

Test Case Description: This test case ensures that the identities of reviewers are hidden from learners during the peer review process.

Test Suite: Peer Review

Test Priority: High

Preconditions:

- Learner is logged into the online learning platform

- Learner is enrolled in a course with active peer review assignments
- Another learner is providing feedback on the learner's assignment

Test Data: N/A

Test Steps:

1. Log into the online learning platform as a learner
2. Navigate to the enrolled course with active peer review assignments
3. Locate the submitted assignment with received feedback
4. Click on the assignment to view details
5. Observe that the identity of the reviewer is hidden or not disclosed in the feedback

Expected Result: The identities of peer reviewers remain hidden, promoting unbiased feedback and fair assessment.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 59

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Certificates and Badges

Feature description -

Upon successful completion of courses or specific milestones, learners receive certificates and badges to recognize their achievements and accomplishments. These credentials provide tangible recognition of learners' efforts, skills, and expertise, enhancing their credentials and demonstrating their commitment to continuous learning and professional development.

Userstory title -

As a learner, I want to earn certificates and badges so that I can showcase my achievements and skills.

Acceptance criteria -

['Learners earn certificates upon course completion.', 'Badges are awarded for specific achievements and milestones.', 'Certificates and badges are accessible through learner profiles.', 'Learners can share their certificates and badges on social media.', 'Instructors can issue custom badges for special achievements.']

Generated Test Cases:

Test Case Creation Template:

Test Case ID: TC001_CertificatesAndBadges

Test Case Title: Verify that learners earn certificates upon course completion.

Test Case Description: This test case validates if learners receive certificates after finishing a course.

Test Suite: Certificates and Badges

Test Priority: High

Preconditions:

- A learner is enrolled in a course.
- The learner has completed all modules and quizzes within the course.
- The course has been set up to award a certificate upon completion.

Test Data: Learner account credentials and a completed course.

Test Steps:

1. Log in as a learner.
2. Access the list of enrolled courses.
3. Select the completed course from the list.
4. Check for a certificate download link or icon on the course completion page.
5. Click the download link or icon to download the certificate.

Postconditions:

- The certificate has been downloaded successfully.
- The certificate displays the course name, learner's name, completion date, and any other relevant details.

Expected Result: Upon course completion, a learner is able to download a certificate that includes the appropriate information.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002_SpecificAchievementBadges

Test Case Title: Verify that badges are awarded for specific achievements and milestones.

Test Case Description: This test case verifies if badges are correctly awarded for specific achievements and milestones.

Test Suite: Certificates and Badges

Test Priority: High

Preconditions:

- A learner is enrolled in a course.

- The learner has completed prerequisite modules or quizzes within the course.
- The course has been set up to award badges for specific achievements.

Test Data: Learner account credentials and a course with specific badge milestones.

Test Steps:

1. Log in as a learner.
2. Access the list of enrolled courses.
3. Select a course with specific badge milestones from the list.
4. Complete the required modules or quizzes to earn the badge.
5. Check for the badge on the learner's profile page.

Postconditions:

- The badge has been correctly awarded and displayed on the learner's profile page.

Expected Result: Upon meeting specific milestones, learners receive and can view badges on their profile pages.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003_AccessCertificatesAndBadges

Test Case Title: Verify that certificates and badges are accessible through learner profiles.

Test Case Description: This test case ensures that learners can easily access their certificates and badges through their profiles.

Test Suite: Certificates and Badges

Test Priority: Medium

Preconditions:

- A learner has completed at least one course and received a certificate and/or badge.
- The learner is logged in.

Test Data: Learner account credentials and a completed course.

Test Steps:

1. Log in as a learner.
2. Access the profile page.
3. Check for the "Certificates" and "Badges" sections on the profile page.
4. Click on each section to view the corresponding certificates and badges.

Postconditions:

- Certificates and badges are displayed by course or achievement.

Expected Result: Learners can view their certificates and badges by accessing their profiles and clicking on the corresponding sections.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004_ShareCertificatesSocialMedia

Test Case Title: Verify that learners can share their certificates and badges on social media.

Test Case Description: This test case checks if learners can share their certificates and badges on popular social media platforms.

Test Suite: Certificates and Badges

Test Priority: Low

Preconditions:

- A learner has completed at least one course and received a certificate and/or badge.
- The learner is logged in.
- The learner has a social media account.

Test Data: Learner account credentials, a completed course, and social media account credentials.

Test Steps:

1. Log in as a learner.
2. Access the profile page.
3. Check for the "Certificates" and "Badges" sections on the profile page.
4. Select a certificate or badge to view.
5. Click the share button and choose a social media platform.
6. Log in to the social media account.
7. Post the certificate or badge.

Postconditions:

- The certificate or badge is posted on the social media platform.

Expected Result: Learners can share their certificates and badges on social media by clicking the share button and logging into their social media accounts.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005_CustomBadges

Test Case Title: Verify that instructors can issue custom badges for special achievements.

Test Case Description: This test case validates if instructors can issue custom badges for special achievements.

Test Suite: Certificates and Badges

Test Priority: Low

Preconditions:

- An instructor has created a custom badge for a special achievement.
- Learner(s) have met the requirements for the custom badge.

Test Data: Instructor account credentials, learner account credentials, and a custom badge.

Test Steps:

1. Log in as an instructor.
2. Access the course with the custom badge.
3. Verify the custom badge is available for issue.
4. Issue the custom badge to the learner who has met the requirements.
5. Log in as the learner.
6. Check for the custom badge on the learner's profile page.

Postconditions:

- The custom badge has been correctly awarded and displayed on the learner's profile page.

Expected Result: Instructors can create and issue custom badges for special achievements, which learners can view on their profile pages.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Data - 60

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Gamification Elements

Feature description -

Gamification elements such as points, levels, and leaderboards are strategically integrated into the learning experience to motivate learners and foster engagement. By incorporating game-like mechanics and incentives, the platform incentivizes active participation, goal attainment, and healthy competition, enhancing learner motivation and enjoyment.

Userstory title -

As a learner, I want gamification elements so that I can be motivated to achieve my learning goals.

Acceptance criteria -

['Learners earn points for completing activities.', 'Levels are achieved based on accumulated points.', 'Leaderboards display top-performing learners.', 'Badges are awarded for reaching milestones.', 'Learners can view their progress in a gamification dashboard.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC001_GamificationElements_Points

Test Case Title: Verify that learners earn points for completing activities.

Test Suite: Gamification Elements

Test Priority: High

Preconditions:

- User is logged in.
- User has enrolled in a course.

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Enroll in a course.
3. Complete the first lecture of the course.
4. Mark the lecture as complete.
5. Verify that the user's point balance has increased.

Postconditions:

- User's point balance is updated.
- User's progress in the course is tracked.

Expected Result: User earns points for completing lectures, and user's point balance is accurately reflected in the system.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC002_GamificationElements_Levels

Test Case Title: Verify that levels are achieved based on accumulated points.

Test Suite: Gamification Elements

Test Priority: High

Preconditions:

- User is logged in.
- User has enrolled in a course.
- User has earned a sufficient number of points.

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Enroll in a course.
3. Complete lectures and other activities to earn enough points.
4. Verify that the user's level has increased in the gamification dashboard.

Postconditions:

- User's level is updated based on accumulated points.

Expected Result: User advances levels based on a predefined threshold of accumulated points.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC003_GamificationElements_Leaderboards

Test Case Title: Verify that leaderboards display top-performing learners.

Test Suite: Gamification Elements

Test Priority: Medium

Preconditions:

- User is logged in.
- User has enrolled in a course.

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Enroll in a course.
3. Complete lectures and other activities to earn points.
4. Verify that the user's position on the leaderboard is accurate compared to other learners.

Postconditions:

- User's position is accurately reflected on the leaderboard.

Expected Result: Top learners are ranked on the leaderboard based on accumulated points.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC004_GamificationElements_Badges

Test Case Title: Verify that badges are awarded for reaching milestones.

Test Suite: Gamification Elements

Test Priority: Medium

Preconditions:

- User is logged in.
- User has enrolled in a course.

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Enroll in a course.
3. Complete lectures, quizzes, and other activities that have badges associated with them.
4. Verify that the user has received the corresponding badges.

Postconditions:

- User's badge collection has been updated.

Expected Result: Users are awarded badges for achieving milestones in courses.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC005_GamificationElements_Dashboard

Test Case Title: Verify that learners can view their progress in a gamification dashboard.

Test Suite: Gamification Elements

Test Priority: High

Preconditions:

- User is logged in.
- User has enrolled in a course.

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Enroll in a course.
3. Complete lectures, quizzes, and other activities.
4. Access the gamification dashboard.
5. Verify that the user's total points, levels, badges, and leaderboard position are visible.

Postconditions:

- User's gamification progress is clearly displayed in the dashboard.

Expected Result: Users can view their progress in a gamification dashboard.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 61

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Multilingual Support

Feature description -

The platform offers robust multilingual support, catering to learners from diverse linguistic backgrounds and facilitating access to educational resources in their preferred language. Through multilingual interfaces, course materials, and support services, the platform ensures inclusivity, accessibility, and cultural relevance for all learners.

Userstory title -

As a learner, I want multilingual support so that I can access the platform in my preferred language.

Acceptance criteria -

['The platform interface is available in multiple languages.', 'Course materials are available in different languages.', 'Learners can select their preferred language at any time.', 'Multilingual support includes translated help and support resources.', 'Instructors can provide content in multiple languages.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC-MLS-001

Test Case Title: Verify that the platform interface is available in multiple languages

Test Case Description: This test case verifies that the user can access the platform's interface in different languages.

Test Suite: Multilingual Support

Test Priority: High

Preconditions:

- User account created
- User is logged in
- Multiple language options are available in settings

Test Data: French, Spanish, German, and Mandarin

Test Steps:

1. Log in as a user
2. Navigate to settings
3. Select a language (e.g. French) from the list of available languages
4. Confirm the language change by checking for language-specific text on the interface
5. Repeat steps 3-4 for other languages (e.g. Spanish, German, Mandarin)

Postconditions:

- The platform interface displays the selected language
- The user remains logged in

Test Expected Result: The platform interface displays the chosen language accurately.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-MLS-002

Test Case Title: Verify that course materials are available in different languages

Test Case Description: This test case validates that course materials are translated and available in different languages.

Test Suite: Multilingual Support

Test Priority: High

Preconditions: User is logged in and enrolled in a course

Test Data: French, Spanish, German, and Mandarin

Test Steps:

1. Access a course as a user
2. Navigate to the course materials section
3. Select a language (e.g. French) from the list of available languages
4. Check for language-specific course content (e.g. video subtitles, text)
5. Repeat steps 3-4 for other languages (e.g. Spanish, German, Mandarin)

Postconditions:

- Course materials are available in the selected language

Test Expected Result: Course materials, including videos and text, are accurately translated into the chosen language.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-MLS-003

Test Case Title: Verify that learners can select their preferred language at any time

Test Case Description: This test case confirms that learners can change their preferred language during their session.

Test Suite: Multilingual Support

Test Priority: Medium

Preconditions: User is logged in and has access to multiple language options

Test Data: No test data needed

Test Steps:

1. Log in as a user
2. Access any page on the platform
3. Change the language preference (e.g. from English to Spanish)
4. Refresh the page

Test Expected Result: The platform interface changes to the newly selected language.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-MLS-004

Test Case Title: Verify that multilingual support includes translated help and support resources

Test Case Description: This test case validates that help and support resources, such as FAQs and guides, are available in different languages.

Test Suite: Multilingual Support

Test Priority: Medium

Preconditions: User is logged in

Test Data: French, Spanish, German, and Mandarin

Test Steps:

1. Access help or support section as a user
2. Check for language-specific resources (e.g. FAQs, articles, guides)
3. Change the language preference (e.g. from English to German)
4. Repeat steps 1-3

Postconditions:

- Help and support resources are available in the selected language

Test Expected Result: Help and support resources are accurately translated into the chosen language.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC-MLS-005

Test Case Title: Verify that instructors can provide content in multiple languages

Test Case Description: This test case confirms that instructors can upload course materials in different languages.

Test Suite: Multilingual Support

Test Priority: Low

Preconditions: Instructor account created and logged in, course created

Test Data: French, Spanish, German, and Mandarin

Test Steps:

1. Log in as an instructor
2. Access a course you have created
3. Upload language-specific course materials (e.g. video subtitles, text)
4. Repeat steps 3 for other languages (e.g. Spanish, German, Mandarin)

Test Expected Result: The instructor can upload course materials in different languages, allowing learners to access the content in their preferred language.

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 62

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Accessibility Features

Feature description -

The platform prioritizes accessibility by incorporating features such as screen reader compatibility, adjustable font sizes, and color contrast options. These accessibility features ensure that learners with disabilities can access and navigate the platform with ease, promoting inclusivity and equitable access to educational opportunities.

Userstory title -

As a learner with disabilities, I want accessibility features so that I can use the platform effectively.

Acceptance criteria -

['The platform is compatible with screen readers.', 'Learners can adjust font sizes for better readability.', 'Color contrast options are available for improved visibility.', 'Accessibility features are easy to activate and customize.', 'Support resources are available to assist learners with disabilities.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that the platform is compatible with screen readers.

Test Case Description: Validate that popular screen readers, such as JAWS and NVDA, can effectively navigate and read aloud interface elements, text, and user inputs on the online learning platform.

Test Suite: Accessibility Features

Test Priority: High

Preconditions:

- * Screen reader software installed

- * User logged in to the platform

Test Data: No test data needed

Test Steps:

1. Launch the online learning platform.
2. Log in using valid credentials.
3. Activate the screen reader software.
4. Navigate the platform using the screen reader.
5. Attempt to access courses, lectures, and quizzes.
6. Attempt to use interactive features like video lectures, downloadable resources, and discussion forums.

Postconditions:

- * Screen reader effectively reads out platform elements, text, and user inputs.

- * User can navigate and use the platform with the screen reader.

Expected Result: The platform is compatible with screen readers, enabling learners with visual impairments to effectively navigate and use the platform.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Compatibility Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that learners can adjust font sizes for better readability.

Test Case Description: Validate that users can increase and decrease font sizes on the platform, improving readability for learners with visual impairments.

Test Suite: Accessibility Features

Test Priority: High

Preconditions:

* User logged in to the platform

Test Data: No test data needed

Test Steps:

1. Launch the online learning platform.
2. Log in using valid credentials.
3. Access any text-based content (course description, lecture notes, etc.).
4. Increase and decrease font sizes using the provided interface options.
5. Verify the readability of the text after adjusting font sizes.

Postconditions:

* Text remains legible and formatted after increasing or decreasing font sizes.

Expected Result: The platform allows learners to easily adjust font sizes, promoting readability and accessibility for users with visual impairments.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that color contrast options are available for improved visibility.

Test Case Description: Validate that users can adjust color contrast settings on the platform, enhancing

visibility for learners with visual impairments.

Test Suite: Accessibility Features

Test Priority: High

Preconditions:

- * User logged in to the platform

Test Data: No test data needed

Test Steps:

1. Launch the online learning platform.
2. Log in using valid credentials.
3. Access the platform settings.
4. Locate and enable color contrast options.
5. Verify the color contrast changes on the platform interface.

Postconditions:

- * Color contrast settings are available and functional.
- * Interface elements, text, and user inputs are visually distinguishable after adjusting color contrast.

Expected Result: The platform provides color contrast options, enabling learners with visual impairments to effectively use the platform.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that accessibility features are easy to activate and customize.

Test Case Description: Validate that users can quickly and easily activate, deactivate, and customize accessibility features on the platform.

Test Suite: Accessibility Features

Test Priority: High

Preconditions:

- * User logged in to the platform

Test Data: No test data needed

Test Steps:

1. Launch the online learning platform.
2. Log in using valid credentials.
3. Locate and activate accessibility features (screen reader, font size adjustment, color contrast).
4. Customize accessibility features as needed.
5. Attempt to deactivate accessibility features.
6. Verify the ease of activation, customization, and deactivation.

Postconditions:

- * Accessibility features are easily accessible and customizable.
- * Users can quickly activate and deactivate accessibility features.

Expected Result: The platform enables learners with disabilities to easily activate, customize, and deactivate accessibility features, improving their overall user experience.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Usability Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that support resources are available to assist learners with disabilities.

Test Case Description: Validate that the platform provides support resources, such as documentation, tutorials, and contact information, to aid learners with disabilities in using the platform effectively.

Test Suite: Accessibility Features

Test Priority: Medium

Preconditions:

* User logged in to the platform

Test Data: No test data needed

Test Steps:

1. Launch the online learning platform.
2. Log in using valid credentials.
3. Locate support resources on the platform (help center, FAQ, contact information).
4. Verify the availability and relevance of support resources for learners with disabilities.

Postconditions:

- * Support resources are easily accessible.
- * Relevant support resources are available for learners with disabilities.

Expected Result: The platform offers support resources that cater to the needs of learners with disabilities, promoting their ability to effectively use and navigate the platform.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Usability Testing

Test Case Approach: Positive

Data - 63

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Discussion Moderation

Feature description -

Instructors actively moderate discussion forums to maintain a conducive learning environment, ensure respectful discourse, and enforce community guidelines. By facilitating constructive interactions and discouraging disruptive behavior, discussion moderation fosters a positive and collaborative learning atmosphere for all participants.

Userstory title -

As an instructor, I want to moderate discussions so that I can maintain a positive and respectful learning environment.

Acceptance criteria -

['Instructors can edit, delete, or highlight posts in discussions.', 'Instructors can mute or ban disruptive users.', 'Moderation actions are logged and reported.', 'Community guidelines are visible to all participants.', 'Notifications are sent for moderated actions.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC-MOD001

Test Case Title: Verify that instructors can edit discussion posts

Test Case Description: As an instructor, verify that you can edit a student's discussion post to maintain clarity and accuracy.

Test Suite: Discussion Moderation

Test Priority: High

Preconditions:

- * Successfully log in as an instructor
- * Navigate to a course with active discussions
- * Locate a student's post in the discussion forum

Test Data: Student's discussion post content

Test Steps:

1. Click on the "Edit" button next to the student's post.
2. Modify the post content with relevant changes.
3. Click the "Save" button to update the post.

Postconditions:

- * The post is updated with the instructor's edits
- * The student is notified of the post edit

Expected Result: Post is successfully edited and updated, and the student receives a notification.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-MOD002

Test Case Title: Verify that instructors can delete discussion posts

Test Case Description: As an instructor, verify that you can delete a student's or your own post to remove irrelevant or inappropriate content.

Test Suite: Discussion Moderation

Test Priority: High

Preconditions:

- * Successfully log in as an instructor
- * Navigate to a course with active discussions
- * Locate a student's or your own post in the discussion forum

Test Data: Student's or instructor's discussion post content

Test Steps:

1. Click on the "Delete" button next to the post.

2. Confirm the deletion action.

Postconditions:

- * The post is removed from the discussion forum
- * The student is notified of the post deletion (if applicable)

Expected Result: Post is successfully deleted, and the appropriate notifications are sent (if applicable).

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-MOD003

Test Case Title: Verify that instructors can highlight posts in discussions

Test Case Description: As an instructor, verify that you can highlight a post to draw attention to important or helpful content.

Test Suite: Discussion Moderation

Test Priority: Medium

Preconditions:

- * Successfully log in as an instructor
- * Navigate to a course with active discussions
- * Locate a student's or instructor's post in the discussion forum

Test Data: Student's or instructor's discussion post content

Test Steps:

1. Click on the "Highlight" button next to the post.
2. Verify that the post background changes to indicate a highlight.

Expected Result: Post is successfully highlighted with a visible background change.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-MOD004

Test Case Title: Verify that instructors can mute or ban disruptive users

Test Case Description: As an instructor, verify that you can mute or ban a user who consistently engages in disruptive behavior.

Test Suite: Discussion Moderation

Test Priority: High

Preconditions:

- * Successfully log in as an instructor
- * Navigate to a course with active discussions
- * Locate a disruptive user's post in the discussion forum

Test Data: Disruptive user's username

Test Steps:

1. Click on the disruptive user's username in the post.
2. Select the "Mute" or "Ban" option from the user's profile.
3. Confirm the mute or ban action.

Postconditions:

- * The disruptive user cannot post or comment in the course discussions
- * The disruptive user is notified of the mute or ban

Expected Result: User is successfully muted or banned, and the appropriate notifications are sent.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC-MOD005

Test Case Title: Verify that moderation actions are logged and reported

Test Case Description: As an instructor, verify that moderation actions are logged for future reference and reported to relevant stakeholders.

Test Suite: Discussion Moderation

Test Priority: High

Preconditions:

- * Successfully log in as an instructor
- * Navigate to a course with active discussions

Test Steps:

1. Moderate a student's discussion post by editing, deleting, or highlighting it.
2. Mute or ban a disruptive user.
3. Navigate to the moderation log page.

Expected Result: All moderation actions are displayed in the log for review and reporting.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 64

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Learning Analytics

Feature description -

Instructors leverage robust learning analytics dashboards to gain insights into learner engagement, performance trends, and course effectiveness. By analyzing data on learner interactions, progress, and outcomes, instructors can make informed decisions, personalize instruction, and optimize the learning experience to meet learners' needs effectively.

Userstory title -

As an instructor, I want access to learning analytics so that I can understand learner engagement and performance.

Acceptance criteria -

['Learning analytics dashboards display engagement metrics.', 'Performance trends are visualized over time.', 'Instructors can access detailed reports on learner progress.', 'Analytics data is updated in real-time.', 'Instructors can export analytics reports for further analysis.']

Generated Test Cases:

Test Case 1:

Test Case ID: LAD-001

Test Case Title: Verify that learning analytics dashboards display engagement metrics.

Test Case Description: Validate that the learning analytics dashboard displays relevant engagement metrics for each course, including the number of active learners, time spent on the platform, and interaction rates.

Test Suite: Learning Analytics Dashboard

Test Priority: High

Preconditions:

- User is logged in as an instructor.
- A course with learners and engagement data is available.

Test Data: No test data needed

Test Steps:

1. Log in as an instructor.
2. Navigate to the learning analytics dashboard.
3. Select a course with learners and engagement data.
4. Review the engagement metrics displayed for the course.

Postconditions:

- Engagement metrics are displayed accurately for the selected course.

Expected Result: Engagement metrics, such as active learners, time spent on the platform, and interaction rates, are displayed accurately for the selected course.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: LAD-002

Test Case Title: Verify that performance trends are visualized over time.

Test Case Description: Ensure that the learning analytics dashboard visualizes performance trends for a selected course, displaying data such as quiz scores, assessment results, and completion rates over a specified time period.

Test Suite: Learning Analytics Dashboard

Test Priority: High

Preconditions:

- User is logged in as an instructor.
- A course with learners, performance data, and a specified time period is available.

Test Data: No test data needed

Test Steps:

1. Log in as an instructor.
2. Navigate to the learning analytics dashboard.
3. Select a course with learners, performance data, and a specified time period.
4. Review the performance trends visualized for the course over the specified time period.

Postconditions:

- Performance trends are visualized accurately for the selected course and time period.

Expected Result: Performance trends, such as quiz scores, assessment results, and completion rates, are visualized accurately for the selected course and time period.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: LAD-003

Test Case Title: Verify that instructors can access detailed reports on learner progress.

Test Case Description: Validate that the learning analytics dashboard provides detailed reports on learner progress, including individual learner scores, completion rates, and time spent on the learning platform.

Test Suite: Learning Analytics Dashboard

Test Priority: High

Preconditions:

- User is logged in as an instructor.
- A course with learners and progress data is available.

Test Data: No test data needed

Test Steps:

1. Log in as an instructor.
2. Navigate to the learning analytics dashboard.
3. Select a course with learners and progress data.
4. Review the detailed reports on learner progress, including individual learner scores, completion rates, and time spent on the learning platform.

Postconditions:

- Detailed reports on learner progress are displayed accurately for the selected course.

Expected Result: Detailed reports on learner progress, including individual learner scores, completion rates, and time spent on the learning platform, are displayed accurately for the selected course.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: LAD-004

Test Case Title: Verify that analytics data is updated in real-time.

Test Case Description: Validate that the learning analytics dashboard updates engagement metrics, performance trends, and learner progress data in real-time, accurately reflecting the current state of a course.

Test Suite: Learning Analytics Dashboard

Test Priority: High

Preconditions:

- User is logged in as an instructor.
- A course with learners, engagement data, and performance data is available.

Test Data: No test data needed

Test Steps:

1. Log in as an instructor.
2. Navigate to the learning analytics dashboard.
- 3.

```diff

- Select a course with learners, engagement data, and performance data.
4. Perform an action that should impact the real-time data, such as having a learner complete a module or take an assessment.
  5. Refresh the dashboard and verify that the displayed data reflects the recent action performed.

```

Postconditions:

- Analytics data is updated accurately in real-time.

Expected Result: Analytics data, including engagement metrics, performance trends, and learner progress, is updated accurately in real-time to reflect recent actions performed.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: LAD-005

Test Case Title: Verify that instructors can export analytics reports for further analysis.

Test Case Description: Ensure that the learning analytics dashboard allows instructors to export analytics reports as a CSV or Excel file, enabling further analysis and sharing of insights with stakeholders.

Test Suite: Learning Analytics Dashboard

Test Priority: Medium

Preconditions:

- User is logged in as an instructor.
- A course with learners, engagement data, and performance data is available.

Test Data: No test data needed

Test Steps:

1. Log in as an instructor.
2. Navigate to the learning analytics dashboard.
3. Select a course with learners, engagement data, and performance data.
4. Export the analytics report as a CSV or Excel file.
5. Verify that the exported file contains accurate data, including engagement metrics, performance trends, and learner progress.

Postconditions:

- Analytics report is exported accurately as a CSV or Excel file.

Expected Result: Analytics report, including engagement metrics, performance trends, and learner progress, is exported accurately as a CSV or Excel file for further analysis.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Social Learning Features

Feature description -

The platform integrates social learning features such as group projects, peer collaboration tools, and social sharing functionalities to foster community engagement and knowledge exchange. By facilitating interaction, networking, and collaboration among learners, social learning features enrich the learning experience and promote peer-to-peer support and mentorship.

Userstory title -

As a learner, I want social learning features so that I can collaborate with peers and engage in group projects.

Acceptance criteria -

['Learners can create and join groups for projects.', 'Collaboration tools are available for group communication.', 'Learners can share resources and feedback within groups.', 'Instructors can assign group projects and monitor progress.', 'Social sharing features allow learners to share achievements.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC-SLF-001

Test Case Title: Verify that learners can create groups for projects

Test Case Description: As a learner, I want to create a new group for a project so that I can collaborate with peers. This test case covers the ability of learners to create groups.

Test Suite: Social Learning Features

Test Priority: High

Preconditions:

- User is logged in
- User has access to the group creation feature

Test Data: Project name, group description, invitee email addresses

Test Steps:

1. Log in as a learner
2. Navigate to the group creation page
3. Enter the project name, group description, and invitee email addresses
4. Click on the 'Create Group' button

Postconditions:

- Group is created
- Invited learners receive group invitations

Expected Result: The group is successfully created, and invited learners can accept the group invitation.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-SLF-002

Test Case Title: Verify that collaboration tools are available for group communication

Test Case Description: As a learner, I want to use collaboration tools to communicate with group members so that we can discuss our project. This test case covers the availability and functionality of group communication tools.

Test Suite: Social Learning Features

Test Priority: Medium

Preconditions:

- User is a member of a group
- Group has collaboration tools enabled

Test Data: Test messages

Test Steps:

1. Log in as a learner
2. Navigate to the group page
3. Use collaboration tools (e.g., chat, forum) to send a test message
4. Verify that the message is delivered and displayed correctly

Postconditions:

- Test message is visible and accessible to all group members

Expected Result: Collaboration tools work properly, allowing learners to send and receive messages within the group.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-SLF-003

Test Case Title: Verify that learners can share resources and feedback within groups

Test Case Description: As a learner, I want to share resources and feedback with group members so that we can enhance our learning experience. This test case covers the ability to share resources and provide feedback within groups.

Test Suite: Social Learning Features

Test Priority: Medium

Preconditions:

- User is a member of a group

Test Data: Resource files, feedback comments

Test Steps:

1. Log in as a learner
2. Navigate to the group page
3. Upload resource files or provide feedback comments
4. Verify that the uploaded files and comments are visible and accessible to all group members

Postconditions:

- Shared resources are available on the group page
- Feedback comments are visible and associated with the appropriate user

Expected Result: Learners can successfully share resources and provide feedback within groups.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-SLF-004

Test Case Title: Verify that instructors can assign group projects and monitor progress

Test Case Description: As an instructor, I want to create and assign group projects so that learners can collaborate and practice their skills. This test case covers the ability of instructors to create, assign, and monitor group projects.

Test Suite: Social Learning Features

Test Priority: High

Preconditions:

- Instructor is logged in
- Instructor has access to the group project creation feature

Test Data: Project name, project description, due date, group members

Test Steps:

1. Log in as an instructor
2. Navigate to the group project creation page
3. Enter the project name, description, due date, and select group members
4. Click on the 'Create Group Project' button
5. Access the group to monitor the progress of the group project
6. Verify that the project and progress are visible and editable by the instructor

Postconditions:

- Group project is created
- Instructor has access to monitor and edit the group project

Expected Result: Instructors can successfully create and assign group projects, as well as monitor the progress of group projects.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC-SLF-005

Test Case Title: Verify that social sharing features allow learners to share achievements

Test Case Description: As a learner, I want to share my achievements (e.g., completing a course, passing a

quiz) with my network so that I can showcase my accomplishments. This test case covers the ability to socially share achievements.

Test Suite: Social Learning Features

Test Priority: Low

Preconditions:

- User is logged in
- User has completed a course or passed a quiz

Test Data: Social media account information, achievement details

Test

Data - 66

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Content Authoring Tools

Feature description -

Instructors utilize powerful content authoring tools to create and publish course content, including multimedia presentations, interactive modules, and rich media resources. These authoring tools empower instructors to design engaging and interactive learning materials that cater to diverse learning styles and preferences, enhancing the effectiveness and impact of instruction.

Userstory title -

As an instructor, I want powerful content authoring tools so that I can create engaging and interactive course materials.

Acceptance criteria -

['Instructors can create multimedia presentations.', 'Interactive modules can be developed and integrated.', 'Rich media resources, such as videos and animations, are supported.', 'Content authoring tools are user-friendly and intuitive.', 'Instructors can preview and edit content before publishing.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that instructors can create multimedia presentations

Test Case Description: This test case verifies that an instructor can create a multimedia presentation using the content authoring tools.

Test Suite: Content Authoring Tools

Test Priority: High

Preconditions:

- User is logged in as an instructor
- The instructor has access to the content authoring tools

Test Data: A set of images, audio, and video files

Test Steps:

1. Log in as an instructor.
2. Navigate to the content authoring tools.
3. Select the "Create Presentation" option.
4. Upload images, audio, and video files as required.
5. Arrange and format the multimedia elements.
6. Save the presentation.

Postconditions:

- The presentation is saved and accessible in the instructor's content library

Expected Result: A multimedia presentation containing images, audio, and video elements is successfully created.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that interactive modules can be developed and integrated

Test Case Description: This test case verifies that an instructor can create and integrate interactive modules within course content.

Test Suite: Content Authoring Tools

Test Priority: High

Preconditions:

- User is logged in as an instructor
- The instructor has access to the content authoring tools
- A multimedia presentation has been created

Test Data: Interactive content (e.g., quizzes, polls, discussion boards)

Test Steps:

1. Log in as an instructor.
2. Navigate to the content authoring tools.
3. Open an existing multimedia presentation.
4. Select the "Add Interactive Module" option.
5. Choose and configure an interactive content type.
6. Save the presentation with the integrated interactive module.

Postconditions:

- The presentation with the integrated interactive module is saved and accessible in the instructor's content

library

Expected Result: An interactive module (quiz, poll, discussion board) is successfully created and integrated within the multimedia presentation.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that rich media resources, such as videos and animations, are supported

Test Case Description: This test case verifies that an instructor can upload and incorporate rich media resources, specifically videos and animations, within course content.

Test Suite: Content Authoring Tools

Test Priority: Medium

Preconditions:

- User is logged in as an instructor
- The instructor has access to the content authoring tools

Test Data: Video and animation files

Test Steps:

1. Log in as an instructor.
2. Navigate to the content authoring tools.
3. Select the "Create Text-Based Content" option.
4. Upload a video or animation file as needed.
5. Verify that the file is properly embedded within the content.
6. Save and preview the content.

Postconditions:

- The content with the uploaded video or animation file is saved and accessible in the instructor's content

library

Expected Result: The video or animation file is properly embedded within the course content and can be played back.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that content authoring tools are user-friendly and intuitive

Test Case Description: This test case verifies that the content authoring tools can be easily accessed and navigated by instructors with minimal training.

Test Suite: Content Authoring Tools

Test Priority: Low

Preconditions:

- User is logged in as an instructor

Test Steps:

1. Log in as an instructor.
2. Locate and click on the "Content Authoring Tools" link.
3. Verify a user-friendly interface with clear navigation options.
4. Attempt to create course content using the tools.
5. Verify that the created content is saved and accessible.

Expected Result: The content authoring tools interface is user-friendly and intuitive, enabling the instructor to create and save course content with ease.

Severity: Minor

Priority: Low

Type of Testing: Usability Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that instructors can preview and edit content before publishing

Test Case Description: This test case verifies that instructors have the ability to preview and edit course content prior to publishing.

Test Suite: Content Authoring Tools

Test Priority: Medium

Preconditions:

- User is logged in as an instructor
- The instructor has access to the content authoring tools
- Course content (e.g., presentations, text-based content, interactive modules) has been created

Test Data: None

Test Steps:

1. Log in as an instructor.
2. Navigate to the content authoring tools.
3. Open a piece of content for editing.
4. Preview the content and verify accuracy.
5. Edit the content as needed.
6. Save and preview the updated content.

Postconditions:

- The updated content is saved and accessible in the instructor's content library

Expected Result: Instructors can preview and edit course content prior to publishing, ensuring accuracy and completeness.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Data - 67

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Learning Paths and Curriculum

Feature description -

The platform offers structured learning paths and curricula to guide learners through sequential courses, modules, and topics aligned with specific learning objectives. By providing clear pathways and learning trajectories, the platform streamlines the learning process, facilitates skill development, and ensures coherence and continuity in learners' educational journeys.

Userstory title -

As a learner, I want structured learning paths and curricula so that I can achieve my learning objectives in a systematic manner.

Acceptance criteria -

['Learning paths include sequential courses and modules.', 'Curricula are aligned with specific learning objectives.', 'Learners can track their progress within a learning path.', 'Instructors can customize learning paths for different needs.', 'Notifications are sent for upcoming courses or milestones.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC001_LearningPathSequence

Test Case Title: Verify that learning paths include sequential courses and modules.

Test Case Description: This test case validates the sequential arrangement of courses and modules within a learning path.

Test Suite: Learning Path

Test Priority: High

Preconditions:

- User is logged in.
- A learning path with multiple courses and modules exists.

Test Data: Learning path with at least two courses and two modules.

Test Steps:

1. Navigate to the learning path.
2. Verify that courses are displayed in a specific order.
3. Select a course and verify that modules are displayed in a specific order.
4. Repeat steps 2 and 3 for subsequent courses.

Postconditions:

- User remains logged in.
- Learning path, courses, and modules are displayed in the correct sequence.

Expected Result: Learning paths display courses and modules sequentially, in the order specified by the instructor.

Severity: Major

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC002_CurriculumLearningObjectives

Test Case Title: Verify that curricula are aligned with specific learning objectives.

Test Case Description: This test case validates the alignment between curricula and their specified learning objectives.

Test Suite: Curriculum

Test Priority: Medium

Preconditions:

- User is logged in.
- A curriculum with specified learning objectives exists.

Test Data: Curriculum with at least one learning objective.

Test Steps:

1. Navigate to the curriculum.
2. Review the learning objectives listed for the curriculum.
3. Verify that each course within the curriculum supports at least one learning objective.

Postconditions:

- User remains logged in.
- Learning objectives are aligned with the curriculum and its courses.

Expected Result: Curricula and their courses support the specified learning objectives.

Severity: Major

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC003_ProgressTracking

Test Case Title: Verify that learners can track their progress within a learning path.

Test Case Description: This test case validates the learners' ability to track their progress through a learning path.

Test Suite: Learning Path

Test Priority: Medium

Preconditions:

- User is logged in.
- A learning path with multiple courses and modules exists.
- User is enrolled in the learning path.

Test Data: No test data needed.

Test Steps:

1. Navigate to the learning path.
2. Verify that a progress bar or similar tracking feature is displayed.
3. Complete a module within a course.
4. Verify that the progress bar or tracking feature reflects the completed module.
5. Repeat steps 3 and 4 for subsequent modules.

Postconditions:

- User remains logged in.
- Progress tracking feature accurately reflects completed modules.

Expected Result: Learners can track their progress through a learning path, with the tracking feature updating as modules are completed.

Severity: Major

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC004_InstructorCustomization

Test Case Title: Verify that instructors can customize learning paths for different needs.

Test Case Description: This test case validates an instructor's ability to customize a learning path.

Test Suite: Learning Path

Test Priority: Low

Preconditions:

- Instructor is logged in.
- A learning path with multiple courses and modules exists.

Test Data: Learning path with at least two courses and two modules.

Test Steps:

1. Log in as an instructor.
2. Navigate to the learning path.
3. Select "Edit" or a similar option.
4. Modify the sequence of courses or modules.
5. Save and publish the changes.
6. Verify that the learning path displays the updated sequence.

Postconditions:

- Instructor remains logged in.
- Learning path displays the updated sequence.

Expected Result: Instructors can customize learning paths, rearranging courses and modules as needed.

Severity: Minor

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC005_NotificationSystem

Test Case Title: Verify that notifications are sent for upcoming courses or milestones.

Test Case Description: This test case validates the notification system for upcoming courses or milestones.

Test Suite: Notification

Test Priority: Medium

Preconditions:

- User is logged in.

- A learning path with multiple courses and modules exists.

Test Data: Learning path with at least two courses and two modules, with the second course starting within a week.

Test Steps:

1. Navigate to the learning path.
2. Verify that the user has not received a notification regarding the second course.
3. Wait for a week.
4. Verify that the user receives a notification reminding them of the second course.

Postconditions:

- User remains logged in.
- User receives a notification for the second course.

Expected Result: Notifications are sent to users to remind them of upcoming courses or milestones.

Severity: Major

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 68

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Attendance Tracking

Feature description -

Instructors track learner attendance in live classes and webinars to monitor participation, engagement, and accountability. By recording attendance data and participation metrics, instructors can assess learner engagement, identify areas for improvement, and provide targeted support and intervention as needed.

Userstory title -

As an instructor, I want to track learner attendance so that I can monitor participation and engagement.

Acceptance criteria -

['Attendance is automatically recorded during live sessions.', 'Instructors can view attendance reports for each session.', 'Learners can see their attendance records.', 'Attendance data is integrated with participation metrics.', 'Notifications are sent for missed sessions.']

Generated Test Cases:

Test Case ID: TC-AT-001

Test Case Title: Verify that attendance is automatically recorded during live sessions.

Test Case Description: This test case verifies whether the system correctly records attendance for learners participating in live sessions without any manual intervention.

Test Suite: Attendance Tracking

Test Priority: High

Preconditions:

- A live session must be currently in progress.
- At least one learner must be participating in the live session.

Test Data: No test data needed

Test Steps:

1. Log in as an instructor.
2. Schedule and start a live session.
3. Ensure at least one learner joins the live session.
4. Wait for a predetermined period (e.g., 5 minutes) during the live session.

5. End the live session.

Postconditions:

- The attendance record must be generated for the live session.
- The attendance record must include the learner's name, session details, and timestamp.

Expected Result: Attendance is automatically recorded during live sessions for each participating learner.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-AT-002

Test Case Title: Verify that instructors can view attendance reports for each session.

Test Case Description: This test case confirms the ability of instructors to access and review attendance data for all sessions.

Test Suite: Attendance Tracking

Test Priority: High

Preconditions:

- An instructor must be logged in.
- At least one live session with recorded attendance must be available.

Test Data: No test data needed

Test Steps:

1. Log in as an instructor.
2. Navigate to the list of past sessions.
3. Select a session with recorded attendance.
4. Access the attendance report for the selected session.

Postconditions:

- The attendance report must be displayed, including learner names and attendance records.

Expected Result: Instructors can view and access attendance reports for all past sessions with recorded attendance details.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-AT-003

Test Case Title: Verify that learners can see their attendance records.

Test Case Description: This test case ensures that learners can view their participation records for past live sessions.

Test Suite: Attendance Tracking

Test Priority: Medium

Preconditions:

- A learner must be logged in.
- The learner must have attended at least one live session.

Test Data: No test data needed

Test Steps:

1. Log in as a learner.
2. Navigate to the attendance records section.
3. View the list of attended live sessions.

Postconditions:

- The attended live sessions must be displayed.

Expected Result: Learners can see and access their attendance records for past live sessions.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-AT-004

Test Case Title: Verify that notifications are sent for missed sessions.

Test Case Description: This test case confirms that learners receive notifications for sessions they missed.

Test Suite: Attendance Tracking

Test Priority: Medium

Preconditions:

- A learner must be registered for a live session.
- The learner must not attend the live session.

Test Data: No test data needed

Test Steps:

1. Log in as a learner.
2. Register for a future live session.
3. Do not join the live session when it starts.
4. Verify the notification system (e.g., email, in-app) for missed session notifications.

Postconditions:

- A missed session notification must be sent to the learner.

Expected Result: Notifications are sent to learners for missed live sessions.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-AT-005

Test Case Title: Verify that attendance data is integrated with participation metrics.

Test Case Description: This test case confirms that attendance records are integrated with participation

metrics, presenting a holistic view of learner engagement.

Test Suite: Attendance Tracking

Test Priority: Low

Preconditions:

- An instructor must be logged in.
- At least one live session with recorded attendance must be available.

Test Data: No test data needed

Test Steps:

1. Log in as an instructor.
2. Navigate to the list of past sessions.
3. Select a session with recorded attendance.
4. Access the attendance report for the selected session.
5. Review the participation metrics.

Postconditions:

- Attendance records must be integrated with participation metrics, showing engagement levels.

Expected Result: Attendance data is integrated with participation metrics, providing a comprehensive overview of learner engagement.

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 69

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Content Licensing and Rights Management

Feature description -

The platform implements robust content licensing and rights management features to protect intellectual property and ensure compliance with copyright laws. By managing content licenses, usage rights, and permissions, the platform safeguards the interests of content creators and providers, enabling secure and lawful distribution of educational materials.

Userstory title -

As a content creator, I want content licensing and rights management so that my intellectual property is protected and used legally.

Acceptance criteria -

['Content licenses are assigned to each resource.', 'Usage rights and permissions are clearly defined.', 'Content creators can set restrictions on usage and distribution.', 'Instructors and learners are informed of content licensing terms.', 'Notifications are sent for license expiration or renewal.']

Generated Test Cases:

Test Case 1:

Test Case ID: TL-001

Test Case Title: Verify that content licenses are assigned to each resource

Test Suite: Content Management

Test Priority: High

Preconditions:

- User is logged in as a content creator

- A new piece of content has been uploaded

Test Data: Newly uploaded content

Test Steps:

1. Log in as a content creator
2. Upload a new piece of content
3. Check the content details page for assigned license information

Postconditions: Content license is displayed on the content details page

Expected Result: Each resource has a content license assigned to it

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TL-002

Test Case Title: Verify that usage rights and permissions are clearly defined

Test Suite: Content Management

Test Priority: High

Preconditions:

- User is logged in as a content creator
- A piece of content with a content license is available

Test Data: Content with an existing content license

Test Steps:

1. Log in as a content creator
2. View the content details page
3. Check the content license for clearly defined usage rights and permissions

Postconditions: Usage rights and permissions are clearly detailed

Expected Result: Usage rights and permissions for each content license are clearly defined

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TL-003

Test Case Title: Verify that content creators can set restrictions on usage and distribution

Test Suite: Content Management

Test Priority: High

Preconditions:

- User is logged in as a content creator
- A new piece of content is ready for upload

Test Data: Newly uploaded content

Test Steps:

1. Log in as a content creator
2. Upload a new piece of content
3. Set restrictions on usage and distribution for the content license
4. Save the content license

Postconditions: Restrictions are applied to the content license

Expected Result: Content creators can set restrictions on usage and distribution for their content

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TL-004

Test Case Title: Verify that instructors and learners are informed of content licensing terms

Test Suite: Course Management

Test Priority: Medium

Preconditions:

- User is logged in as an instructor or learner
- A course containing licensed content is available

Test Data: Course with licensed content

Test Steps:

1. Log in as an instructor or learner
2. Access the course containing licensed content
3. Check the content details page for content licensing terms

Postconditions: Content licensing terms are displayed

Expected Result: Instructors and learners are informed of content licensing terms

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TL-005

Test Case Title: Verify that notifications are sent for license expiration or renewal

Test Suite: Notification Systems

Test Priority: Medium

Preconditions:

- User is logged in as a content creator

- A piece of content with a content license is available
- Content license has an expiration date

Test Data: Content with an expiring license

Test Steps:

1. Log in as a content creator
2. View the content details page
3. Check for a notification about the upcoming license expiration
4. Wait for the license expiration date
5. Check for a notification about the license expiration
6. Renew the content license
7. Check for a notification about the successful license renewal

Postconditions: Notifications are received for license expiration and renewal

Expected Result: Users receive notifications for license expiration and renewal

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Negative, Positive

Data - 70

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Content Versioning

Feature description -

The platform implements content versioning features, allowing instructors to manage and track multiple versions of course materials. By providing version control, the platform ensures that updates, revisions, and changes are systematically recorded, enabling instructors to maintain a history of content evolution and easily revert to previous versions if needed.

Userstory title -

As an instructor, I want content versioning features so that I can manage and track changes to my course materials.

Acceptance criteria -

['Instructors can create and save multiple versions of content.', 'Version history is maintained and accessible.', 'Instructors can compare different versions of content.', 'Previous versions can be restored if needed.', 'Notifications are sent for significant content updates.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC001_ContentVersioning_CreateVersion

Test Case Title: Verify that instructors can create and save multiple versions of content.

Test Case Description: This test case aims to validate that an instructor can create and save multiple versions of course material.

Test Suite: Content Versioning

Test Priority: High

Preconditions:

- User is logged in as an instructor.
- The instructor has access to create and edit course content.

Test Data: Course material for which versions need to be created.

Test Steps:

1. Navigate to the course content editing section.
2. Create a new version of the existing content by clicking on the "Create New Version" button.
3. Modify the content as required.
4. Save the new version.

Postconditions:

- A new version of the content is saved.
- The previous version remains accessible.

Expected Result: A new version of the content is created, and both versions are accessible and displayed correctly.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC002_ContentVersioning_VersionHistory

Test Case Title: Verify that version history is maintained and accessible.

Test Case Description: This test case aims to ensure that the platform maintains a history of content versions and that instructors can access it.

Test Suite: Content Versioning

Test Priority: High

Preconditions:

- User is logged in as an instructor.
- The instructor has access to course content with at least two versions.

Test Data: No test data needed

Test Steps:

1. Navigate to the course content section.
2. Locate the version history link or button.
3. Click on the version history link or button.

Postconditions:

- A list of versions is displayed.

Expected Result: All versions of the content are displayed in chronological order with the option to view and revert to a previous version.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC003_ContentVersioning_CompareVersions

Test Case Title: Verify that instructors can compare different versions of content.

Test Case Description: This test case aims to validate that instructors can compare different versions of content side-by-side.

Test Suite: Content Versioning

Test Priority: Medium

Preconditions:

- User is logged in as an instructor.
- The instructor has access to course content with at least two versions.

Test Data: No test data needed

Test Steps:

1. Navigate to the course content section.
2. Locate the version history link or button.
3. Select two versions from the list and compare them.

Postconditions:

- The differences between the two versions are highlighted.

Expected Result: The platform highlights the differences between the two selected versions of the content, enabling easy comparison.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC004_ContentVersioning_RestoreVersion

Test Case Title: Verify that previous versions can be restored if needed.

Test Case Description: This test case aims to validate that an instructor can restore a previous version of the course content.

Test Suite: Content Versioning

Test Priority: Medium

Preconditions:

- User is logged in as an instructor.
- The instructor has access to course content with at least two versions.

Test Data: No test data needed

Test Steps:

1. Navigate to the course content section.
2. Locate the version history link or button.

3. Select a previous version from the list and click the "Restore" button.

4. Confirm the action in the pop-up window.

Postconditions:

- The selected version is restored.
- The latest version is saved as a new version in the history.

Expected Result: The selected version of the content is restored, and it becomes the latest version with a new version created in the history.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC005_ContentVersioning_Notifications

Test Case Title: Verify that notifications are sent for significant content updates.

Test Case Description: This test case aims to ensure that instructors receive notifications for significant content updates.

Test Suite: Content Versioning

Test Priority: Low

Preconditions:

- User is logged in as an instructor.
- The instructor has access to course content.

Test Data: A significant update to the course content (e.g., deletion of a module or a major change).

Test Steps:

1. Log in as an instructor.
2. Perform a significant update to the course content.

3. Wait for the notification.

Postconditions:

- A notification is received.

Expected Result: The instructor receives a notification regarding the significant content update, either via email, in-platform notification, or both.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 71

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

API Integration

Feature description -

The platform supports API integration, allowing third-party applications to interact with the platform's data and functionalities. By offering comprehensive API documentation and secure access protocols, the platform enables seamless data exchange and integration with other educational tools and systems.

Userstory title -

As a developer, I want API integration so that I can connect third-party applications to the platform's data and

features.

Acceptance criteria -

['APIs are available for key platform functionalities.', 'Comprehensive API documentation is provided.', 'Secure access protocols are implemented.', 'Third-party applications can authenticate and access APIs.', 'Rate limiting and monitoring features are in place.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC001_API_Integration

Test Case Title: Verify that APIs are available for key platform functionalities

Test Case Description: This test case aims to validate the existence of APIs for essential platform features, ensuring that third-party applications can access and interact with the platform's data and functionalities.

Test Suite: API Integration

Test Priority: High

Preconditions:

- Online Learning Platform is installed and functional
- Third-party application is ready for integration

Test Data: N/A

Test Steps:

1. Identify key platform functionalities
2. Check for available APIs in the platform documentation
3. Confirm that each functionality has a corresponding API

Postconditions: N/A

Expected Result: All key platform functionalities have corresponding APIs in the documentation.

Severity: Critical

Priority: High

Type of Testing: API Testing, Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC002_API_Documentation

Test Case Title: Verify that comprehensive API documentation is provided

Test Case Description: This test case aims to ensure that the platform's API documentation is detailed and includes all required information, enabling third-party applications to authenticate and access APIs with ease.

Test Suite: API Integration

Test Priority: High

Preconditions:

- Online Learning Platform is installed and functional

Test Data: N/A

Test Steps:

1. Access the API documentation
2. Review the documentation for each API
3. Verify that all required elements (endpoint URL, request format, response format, error handling, authentication method) are present for each API

Postconditions: N/A

Expected Result: Complete and detailed API documentation is available for all platform functionalities.

Severity: Major

Priority: High

Type of Testing: API Testing, Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC003_API_Authentication

Test Case Title: Verify that third-party applications can authenticate and access APIs

Test Case Description: This test case aims to validate that third-party applications can authenticate with the platform's APIs using the provided access protocols, enabling secure data exchange.

Test Suite: API Integration

Test Priority: High

Preconditions:

- Online Learning Platform is installed and functional
- Third-party application is ready for integration
- API documentation is available

Test Data: Test user credentials and API keys

Test Steps:

1. Obtain test user credentials and API keys
2. Attempt to authenticate with the platform's APIs using the provided authentication method
3. Verify successful authentication and data access

Postconditions: N/A

Expected Result: Third-party applications can authenticate and access the platform's APIs using the provided authentication method.

Severity: Major

Priority: High

Type of Testing: API Testing, Functional Testing, Security Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC004_API_Rate_Limiting

Test Case Title: Verify that rate limiting and monitoring features are in place

Test Case Description: This test case aims to validate that rate limiting and monitoring features are implemented, ensuring stable and secure API usage for third-party applications.

Test Suite: API Integration

Test Priority: Medium

Preconditions:

- Online Learning Platform is installed and functional
- Third-party application is ready for integration
- API documentation is available

Test Data: Test API requests

Test Steps:

1. Send multiple API requests within a defined time period
2. Monitor response status and latency
3. Verify that a limit is enforced on the number of requests per time period

Postconditions: N/A

Expected Result: Rate limiting and monitoring features are properly implemented for API usage.

Severity: Major

Priority: Medium

Type of Testing: API Testing, Non-Functional Testing, Performance Testing, Load Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC005_API_Negative_Authentication

Test Case Title: Verify that the system handles negative authentication attempts securely

Test Case Description: This test case aims to validate that the platform responds securely to negative authentication attempts, preventing unauthorized access.

Test Suite: API Integration

Test Priority: Medium

Preconditions:

- Online Learning Platform is installed and functional
- API documentation is available

Test Data: Invalid test user credentials and API keys

Test Steps:

1. Obtain invalid test user credentials and API keys
2. Attempt to authenticate with the platform's APIs using the provided authentication method
3. Verify the platform's response indicates unsuccessful authentication

Postconditions: N/A

Expected Result: The platform responds securely to negative authentication attempts by indicating unsuccessful authentication.

Severity: Major

Priority: Medium

Type of Testing: API Testing, Functional Testing, Security Testing

Test Case Approach: Negative

Data - 72

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Data Analytics and Insights

Feature description -

The platform provides data analytics and insights features, enabling instructors and administrators to analyze learner performance, engagement, and outcomes. By leveraging data visualization tools and reporting features,

the platform helps stakeholders make informed decisions to enhance the learning experience.

Userstory title -

As an administrator, I want data analytics and insights so that I can analyze learner performance and improve the platform's effectiveness.

Acceptance criteria -

['Data visualization tools are available for analysis.', 'Reports can be generated for learner performance metrics.', 'Real-time analytics provide insights into engagement and outcomes.', 'Data can be exported for further analysis.', 'Notifications are sent for significant data trends or anomalies.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC-001

Test Case Title: Verify that data visualization tools are available for analysis

Test Case Description: As an administrator, verify that data visualization tools are accessible and functional on the online learning platform.

Test Suite: Data Analytics and Insights

Test Priority: High

Preconditions:

- User is logged in as an administrator
- User has access to the Data Analytics and Insights feature

Test Data: No test data needed

Test Steps:

1. Log in as an administrator.
2. Navigate to the Data Analytics and Insights feature.
3. Verify the presence of data visualization tools, such as charts, graphs, or maps.
4. Interact with the visualization tools to ensure functionality.

Postconditions:

- Data visualization tools have been verified

Expected Result: Data visualization tools are accessible, functional, and provide accurate representations of learner data.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-002

Test Case Title: Verify that reports can be generated for learner performance metrics

Test Case Description: As an administrator, verify that the system generates accurate and complete reports for various learner performance metrics.

Test Suite: Data Analytics and Insights

Test Priority: High

Preconditions:

- User is logged in as an administrator
- User has access to the Data Analytics and Insights feature
- Learner data is available in the system

Test Data: Prepopulated learner data

Test Steps:

1. Log in as an administrator.
2. Navigate to the Data Analytics and Insights feature.
3. Select a learner performance metric (e.g., quiz scores, attendance).
4. Generate a report for the selected metric.
5. Verify that the report includes accurate and complete data for the selected metric.

Postconditions:

- Learner performance report has been generated

Expected Result: The system generates accurate and complete reports for various learner performance metrics.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-003

Test Case Title: Verify that real-time analytics provide insights into engagement and outcomes

Test Case Description: As an administrator, verify that real-time analytics accurately reflect learner engagement and outcomes.

Test Suite: Data Analytics and Insights

Test Priority: High

Preconditions:

- User is logged in as an administrator
- User has access to the Data Analytics and Insights feature
- Learner engagement and outcome data is available in real-time

Test Data: No test data needed

Test Steps:

1. Log in as an administrator.
2. Navigate to the Data Analytics and Insights feature.
3. Access real-time analytics for learner engagement (e.g., time spent on course, discussion board activity).
4. Access real-time analytics for learner outcomes (e.g., quiz scores, course completion status).
5. Verify that real-time analytics accurately represent engagement and outcomes.

Postconditions:

- Real-time analytics have been verified

Expected Result: Real-time analytics provide accurate insights into learner engagement and outcomes.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-004

Test Case Title: Verify that data can be exported for further analysis

Test Case Description: As an administrator, verify that learner data can be exported in a usable format for further analysis.

Test Suite: Data Analytics and Insights

Test Priority: Medium

Preconditions:

- User is logged in as an administrator
- User has access to the Data Analytics and Insights feature
- Learner data is available in the system

Test Data: Prepopulated learner data

Test Steps:

1. Log in as an administrator.
2. Navigate to the Data Analytics and Insights feature.
3. Select the data to be exported (e.g., quiz scores, attendance).
4. Export the selected data.
5. Verify that the exported data is in a usable format and includes all required information.

Postconditions:

- Learner data has been exported

Expected Result: Learner data can be exported in a usable format for further analysis.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC-005

Test Case Title: Verify that notifications are sent for significant data trends or anomalies

Test Case Description: As an administrator, verify that the system sends notifications for significant data trends or anomalies.

Test Suite: Data Analytics and Insights

Test Priority: Medium

Preconditions:

- User is logged in as an administrator
- User has access to the Data Analytics and Insights feature
- System is configured to send notifications for significant data trends or anomalies

Test Data: Preconfigured data trend or anomaly

Test Steps:

1. Log in as an administrator.
2. Navigate to the Data Analytics and Insights feature.
3. Trigger a preconfigured data trend or anomaly.
4. Verify that a notification is received via the preferred method (e.g., email, in-platform notification).

Postconditions:

- Notification has been received for the triggered data trend or anomaly

Expected Result: The system sends notifications for significant data trends or anomalies.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 73

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

User Roles and Permissions

Feature description -

The platform implements a user roles and permissions system, allowing administrators to define and manage different levels of access and functionality for various user groups. This feature ensures that users have appropriate permissions based on their roles, such as instructors, students, and administrators, enhancing security and user experience.

Userstory title -

As an administrator, I want to manage user roles and permissions so that I can control access to platform features.

Acceptance criteria -

['Administrators can create and assign user roles.', 'Permissions are customizable for each role.', 'Users are restricted to functionalities based on their roles.', 'Changes to roles and permissions are logged.', 'Users are notified of their assigned roles and permissions.']

Generated Test Cases:

Test Case 1:

Test Case ID: TL-001

Test Case Title: Verify that administrators can create user roles

Test Case Description: As an administrator, I should be able to create new user roles with customizable permissions.

Test Suite: User Roles Management

Test Priority: High

Preconditions:

- Log in as an administrator
- Be on the "User Roles" management page

Test Data: New role name, role description, and permissions

Test Steps:

1. Click the "Create New Role" button.
2. Enter the new role name and description.
3. Select or deselect the desired permissions for this role.
4. Click the "Save Role" button.

Postconditions:

- New role is created
- New role appears in the list of roles

Expected Result: A new role is successfully created with the assigned permissions.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TL-002

Test Case Title: Verify that users are restricted to functionalities based on their roles

Test Case Description: As a user, I should only have access to features and functionalities allowed by my user role.

Test Suite: Role-Based Access Control

Test Priority: High

Preconditions:

- Log in as a student, instructor, or administrator
- Verify the user's role

Test Data: Accessing various platform features

Test Steps:

1. Attempt to access features as a user with a specific role.
2. Verify whether the user is allowed or denied access.
3. Repeat steps 1-2 for different roles and features.

Postconditions:

- Access is granted or denied based on user role

Expected Result: Users are restricted to functionalities based on their roles, as defined by the administrator.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TL-003

Test Case Title: Verify that changes to roles and permissions are logged

Test Case Description: As an administrator, I should be able to view a log of role and permission changes.

Test Suite: Change Logging

Test Priority: Medium

Preconditions:

- Log in as an administrator
- Make changes to user roles or permissions

Test Data: Changes to roles or permissions

Test Steps:

1. Navigate to the change log.
2. Verify that all changes to roles or permissions are recorded.
3. Check the log details, including the user who made the change, the time of the change, and the nature of the change.

Postconditions:

- Change log is displayed
- Changes are correctly recorded

Expected Result: All changes to roles and permissions are accurately logged, and the logs can be accessed by administrators.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TL-004

Test Case Title: Verify that users are notified of their assigned roles and permissions

Test Case Description: As a user, I should receive a notification when my role or permissions are changed.

Test Suite: User Notification

Test Priority: Medium

Preconditions:

- Log in as a user

- Have an existing role and permissions
- Administrator changes the user's role or permissions

Test Data: Role or permission change

Test Steps:

1. Receive a notification about the role or permission change.
2. Check the notification content for accuracy, including new role, permissions, and the user who made the change.

Postconditions:

- Notification is received
- Notification contains accurate information

Expected Result: Users are notified of role or permission changes, and the notification contains relevant and accurate information.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TL-005

Test Case Title: Verify that users cannot access features outside their role permissions during negative testing

Test Case Description: As a user, I should not be able to access features that are not allowed by my user role during negative testing.

Test Suite: Role-Based Access Control (Negative Testing)

Test Priority: Medium

Preconditions:

- Log in as a user
- Verify the user's role

Test Data: Attempting to access features outside the user's role permissions

Test Steps:

1. Attempt to access features not allowed by the user role.
2. Verify that the user is denied access.

Postconditions:

- Access is denied for features outside the user's role

Expected Result: Users cannot access features outside their role permissions during negative testing.

Severity: Major

Priority: Medium

Type of Testing: Non-Functional Testing

Test Case Approach: Negative

Data - 74

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

User Feedback and Rating System

Feature description -

The platform includes a user feedback and rating system, allowing learners to provide feedback on courses, instructors, and learning materials. This feature enables continuous improvement by gathering user insights and ratings, helping instructors refine their content and approach.

Userstory title -

As a learner, I want to provide feedback and ratings so that I can share my experience and help improve the platform.

Acceptance criteria -

['Learners can rate courses and instructors on a scale.', 'Feedback forms are available for detailed comments.', 'Instructors can view and respond to feedback.', 'Aggregated ratings are displayed publicly.', 'Anonymous feedback options are available.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that learners can rate courses and instructors on a scale

Test Case Description: As a learner, I want to rate a course or instructor on a scale of 1 to 5 stars to express my satisfaction with the learning experience.

Test Suite: User Feedback and Rating System

Test Priority: High

Preconditions:

- User is logged in
- User has completed at least one lecture in the course
- Course or instructor details page is open

Test Data: Course or instructor details page, user's rating

Test Steps:

1. Log in as a registered user
2. Navigate to the course or instructor details page
3. Locate the rating component
4. Select a rating between 1 and 5 stars
5. Click the 'Submit' button

Postconditions:

- The rating is saved and displayed on the course or instructor details page
- The user cannot change their rating without editing it

Expected Result: The user can rate the course or instructor on a scale of 1 to 5 stars, and the rating is accurately displayed and saved.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that feedback forms are available for detailed comments

Test Case Description: As a learner, I want to provide detailed feedback on a course or instructor through a feedback form, enabling the instructor and platform to better understand my learning experience.

Test Suite: User Feedback and Rating System

Test Priority: High

Preconditions:

- User is logged in
- User has completed at least one lecture in the course
- Course or instructor details page is open

Test Data: Course or instructor details page, user's feedback text

Test Steps:

1. Log in as a registered user
2. Navigate to the course or instructor details page
3. Locate the feedback form
4. Enter detailed feedback in the text area
5. Click the 'Submit' button

Postconditions:

- The feedback is saved and associated with the course or instructor
- The user can view their submitted feedback

Expected Result: The user can submit detailed feedback through the feedback form, and the feedback is accurately saved and associated with the correct course or instructor.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that instructors can view and respond to feedback

Test Case Description: As an instructor, I want to view and respond to feedback from learners, fostering a sense of community and continuous improvement.

Test Suite: User Feedback and Rating System

Test Priority: Medium

Preconditions:

- Instructor is logged in
- Learner has submitted feedback for the instructor's course
- Instructor is on the course feedback page

Test Data: Learner's feedback, instructor's response text

Test Steps:

1. Log in as an instructor
2. Navigate to the course feedback page
3. Locate the feedback from the learner
4. Click the 'Reply' button
5. Enter a response in the text area
6. Click the 'Submit' button

Postconditions:

- The instructor's response is saved and displayed with the feedback

Expected Result: The instructor can view, reply to, and save feedback from learners, with the responses accurately displayed and associated with the correct feedback.

Severity: Critical

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that aggregated ratings are displayed publicly

Test Case Description: As a prospective learner, I want to view aggregated ratings from other learners to help me make informed decisions when selecting courses and instructors.

Test Suite: User Feedback and Rating System

Test Priority: Medium

Preconditions:

- Multiple learners have rated the course or instructor
- Course or instructor details page is open

Test Data: Aggregated ratings (number of ratings and average rating)

Test Steps:

1. Navigate to the course or instructor details page
2. Locate the aggregated ratings section
3. Verify the number of ratings
4. Verify the average rating

Expected Result: The aggregated ratings (number of ratings and average rating) are accurately displayed for the course or instructor.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that anonymous feedback options are available

Test Case Description: As a learner, I want the option to provide anonymous feedback, allowing me to share my honest opinion without fear of repercussions.

Test Suite: User Feedback and Rating System

Test Priority: Low

Preconditions:

- User is logged in
- User has completed at least one lecture in the course
- Course or instructor details page is open

Test Data: Course or instructor details page, user's feedback text (anonymous)

Test Steps:

1. Log in as a registered user
2. Navigate to the course or instructor details page
3. Locate the feedback form
4. Select the 'Anonymous' option
5. Enter detailed feedback in the text area
6. Click the 'Submit' button

Postconditions:

- The feedback is saved and associated with the course or instructor
- The user's identity remains anonymous

Expected Result: The user can submit anonymous feedback through the feedback form, and the feedback is accurately saved and associated with the correct course or instructor while keeping the user's identity hidden.

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 75

Project name -

Online Learning Platform

Project description -

Develop an online learning platform that offers courses, lectures, tutorials, and quizzes on various subjects and topics. The platform should provide interactive features like video lectures, downloadable resources, discussion forums, progress tracking, and certifications upon course completion.

Feature name -

Social Media Integration

Feature description -

The platform supports social media integration, allowing learners and instructors to share their achievements, courses, and updates on social media platforms. This feature promotes engagement and community building, enhancing the visibility of the platform's offerings.

Userstory title -

As a learner, I want to share my achievements on social media so that I can showcase my progress and connect with peers.

Acceptance criteria -

['Learners can connect their social media accounts to the platform.', 'Achievements and course completions can be shared on social media.', 'Privacy settings control the visibility of shared content.', 'Instructors can promote courses and updates through social media.', 'Analytics track the reach and engagement of shared

content.']

Generated Test Cases:

Test Case 1:

Test Case ID: SocialMediaIntegration_001

Test Case Title: Verify that learners can connect their social media accounts to the platform

Test Case Description: This test case verifies that learners can successfully connect their social media accounts (e.g., Facebook, Twitter, LinkedIn) to the online learning platform.

Test Suite: Social Media Integration

Test Priority: High

Preconditions:

1. Learner is registered and logged into the platform
2. Social media accounts are created and accessible

Test Data: Learner's social media account credentials

Test Steps:

1. Access the account settings page within the learning platform
2. Select the option to connect a social media account
3. Enter the social media account credentials
4. Authorize the connection request
5. Verify that the connection is successful and the learner's social media profile is displayed

Postconditions:

1. Social media account is connected to the learning platform
2. Learner can share achievements and course completions through the connected account

Expected Result: The learner can connect their social media account to the learning platform, enabling the sharing of achievements and course completions.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: SocialMediaIntegration_002

Test Case Title: Verify that learners can share their achievements on social media

Test Case Description: This test case verifies that learners can share their achievements, such as course completions, on their connected social media accounts.

Test Suite: Social Media Integration

Test Priority: High

Preconditions:

1. Learner is registered and logged into the platform
2. Social media account is connected to the learning platform

Test Data: Course completion information

Test Steps:

1. Access the learner's completed courses page within the learning platform
2. Select the share button for a completed course
3. Choose the connected social media account as the sharing platform
4. Verify that the achievement is shared correctly, displaying the course title and learner's profile

Expected Result: The learner can share their course completion achievements on their connected social media account.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: SocialMediaIntegration_003

Test Case Title: Verify that learners' privacy settings control the visibility of shared content

Test Case Description: This test case verifies that learners' privacy settings, such as post visibility or account privacy, control the visibility of shared content on social media platforms.

Test Suite: Social Media Integration

Test Priority: Medium

Preconditions:

1. Learner is registered and logged into the platform
2. Social media account is connected to the learning platform
3. Learner has adjusted their privacy settings for shared content

Test Data: Course completion information, adjusted privacy settings

Test Steps:

1. Access the learner's completed courses page within the learning platform
2. Select the share button for a completed course
3. Choose the connected social media account as the sharing platform
4. Verify that the shared content complies with the learner's privacy settings (e.g., visibility, post permissions)

Expected Result: Learners' privacy settings control the visibility of shared content on social media platforms.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: SocialMediaIntegration_004

Test Case Title: Verify that instructors can promote courses and updates through social media

Test Case Description: This test case verifies that instructors can promote their courses and updates via social media platforms, engaging learners and attracting potential new students.

Test Suite: Social Media Integration

Test Priority: Medium

Preconditions:

1. Instructor is registered and logged into the platform
2. Instructor has created at least one course
3. Social media account is connected to the learning platform

Test Data: Course title, instructor's profile

Test Steps:

1. Access the instructor's courses page within the learning platform
2. Select the share or promote button for a course
3. Choose the connected social media account as the sharing platform
4. Verify that the course is shared correctly, displaying the course title and instructor's profile

Expected Result: Instructors can promote their courses and updates on social media platforms, reach and engagement can be tracked through analytics.

Severity: Medium

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: SocialMediaIntegration_005

Test Case Title: Verify that analytics track the reach and engagement of shared content

Test Case Description: This test case validates that the platform's analytics accurately track the reach and engagement of achievements and course updates shared on social media platforms.

Test Suite: Analytics & Social Media Integration

Test Priority: Low

Preconditions:

- 1. Learner or instructor is registered and logged into the platform
- 2. Social media account is connected to the learning platform
- 3. Shared content is available on social media

Test Data: Course completion information, course title, social media engagement data (e.g., likes, shares, comments)

Test Steps:

- 1. Access the analytics section within the learning platform
- 2. Verify that shared content is displayed with associated engagement data
- 3. Filter analytics by time frame and social media platform
- 4. Verify that analytics accurately reflect the reach and engagement of the shared content

Expected Result: The analytics track the reach and engagement of shared content on social media platforms, allowing instructors and learners to monitor the effectiveness of their social media communication.

Severity: Medium

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 76

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Activity Tracking

Feature description -

The app records various activities such as running, walking, cycling, and swimming, capturing data on distance, duration, pace, and calories burned. This feature enables users to monitor their daily physical activity levels and progress towards fitness goals.

Userstory title -

As a user, I want to track my activities so that I can monitor my daily physical activity levels.

Acceptance criteria -

['The app records data on distance, duration, pace, and calories burned.', 'Users can view activity summaries.', 'Data is stored for historical analysis.', 'Activity tracking can be paused and resumed.', 'Users receive notifications for milestones.']

Generated Test Cases:

Test Case ID: FTA-001

Test Case Title: Verify that the app records data on distance, duration, pace, and calories burned during an activity.

Test Suite: Activity Tracking

Test Priority: High

Preconditions:

- User is logged in
- User has permission to track activities

- Device GPS and sensors are enabled

Test Data: User performs a 30-minute walk

Test Steps:

1. Start tracking an activity (walking)
2. Walk for 30 minutes
3. Stop tracking the activity
4. View the activity summary

Postconditions:

- Data on distance, duration, pace, and calories burned is displayed

Expected Result: All required metrics (distance, duration, pace, and calories burned) are accurately recorded and displayed for the activity.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: FTA-002

Test Case Title: Verify that users can view activity summaries.

Test Suite: Activity Tracking

Test Priority: High

Preconditions:

- User is logged in
- User has tracked at least one activity

Test Data: User has tracked a 30-minute walk

Test Steps:

1. Navigate to the activity history page
2. Select the most recent activity

3. View the activity summary

Postconditions:

- All required metrics (distance, duration, pace, and calories burned) are displayed

Expected Result: A comprehensive and accurate summary of the user's most recent activity is displayed, including all required metrics.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: FTA-003

Test Case Title: Verify that data is stored for historical analysis.

Test Suite: Activity Tracking

Test Priority: Medium

Preconditions:

- User is logged in
- User has tracked at least one activity

Test Data: User has tracked a 30-minute walk two days ago

Test Steps:

1. Navigate to the activity history page
2. Select an activity from two days ago
3. View the activity summary

Postconditions:

- All required metrics (distance, duration, pace, and calories burned) are displayed

Expected Result: All historical activity data is accurately stored and available for analysis, including all required metrics.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: FTA-004

Test Case Title: Verify that activity tracking can be paused and resumed.

Test Suite: Activity Tracking

Test Priority: Medium

Preconditions:

- User is logged in
- User has permission to track activities
- Device GPS and sensors are enabled

Test Data: User performs a 30-minute walk with a 5-minute pause

Test Steps:

1. Start tracking an activity (walking)
2. Walk for 25 minutes
3. Pause the activity
4. Wait for 5 minutes
5. Resume the activity
6. Walk for another 5 minutes
7. Stop tracking the activity
8. View the activity summary

Postconditions:

- Data on distance, duration, pace, and calories burned is displayed and includes the pause

Expected Result: The app accurately records the activity with the specified pause, including all required metrics.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: FTA-005

Test Case Title: Verify that users receive notifications for milestones.

Test Suite: Activity Tracking

Test Priority: Low

Preconditions:

- User is logged in
- User has permission to track activities
- Device GPS and sensors are enabled
- User has allowed notifications

Test Data: User has a goal of walking 5,000 steps

Test Steps:

1. Start tracking an activity (walking)
2. Walk for 5,000 steps
3. Observe notifications on the device

Postconditions:

- A notification about reaching the milestone (5,000 steps) is displayed

Expected Result: The app delivers a notification upon reaching a specified milestone.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

GPS Tracking

Feature description -

Integrated GPS tracking allows users to accurately track outdoor activities and routes, providing detailed maps, elevation profiles, and location-based metrics. Users can visualize their workout routes, analyze performance data, and explore new routes for added variety and challenge.

Userstory title -

As a user, I want to use GPS tracking so that I can accurately track my outdoor activities and routes.

Acceptance criteria -

['GPS tracking accurately records route data.', 'Users can view detailed maps and elevation profiles.', 'Performance data includes location-based metrics.', 'Routes can be saved and revisited.', 'Notifications are provided for deviations from planned routes.']

Generated Test Cases:

Test Case 1:

Test Case ID: FTA-GPS-001

Test Case Title: Verify that GPS tracking accurately records route data.

Test Case Description: Validate the accuracy of GPS tracking by comparing it with a known route, ensuring distance, duration, and location-based metrics are correctly recorded.

Test Suite: GPS Functionality

Test Priority: High

Preconditions:

- User is outdoors with a clear view of the sky.
- Device has GPS enabled and permission granted for the app.

Test Data: Known outdoor route (e.g., a park or a running trail)

Test Steps:

1. Start the fitness tracker app.
2. Enable GPS tracking.
3. Begin recording the workout.
4. Follow the known outdoor route.
5. Stop the workout and save the route.
6. Compare the recorded route data with the known route.

Postconditions:

- Route data is saved and accessible.

Expected Result: GPS tracking accurately records route data, including distance, duration, and location-based metrics.

Severity: Major

Priority: High

Type of Testing: Functional Testing, System Testing, Location-Based Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: FTA-GPS-002

Test Case Title: Verify that users can view detailed maps and elevation profiles.

Test Case Description: Users should be able to visualize their workout routes on detailed maps and analyze performance data with elevation profiles.

Test Suite: GPS Functionality

Test Priority: High

Preconditions:

- User has completed a workout with GPS tracking enabled.
- Internet connection is available for map display.

Test Data: Recorded workout data with GPS tracking enabled

Test Steps:

1. Open the fitness tracker app.
2. Navigate to the "History" or "Completed Workouts" section.
3. Select a workout with GPS tracking enabled.
4. Verify that the map displays the correct route.
5. Verify that the elevation profile is displayed and accurate.

Postconditions:

- Map and elevation profile are displayed.

Expected Result: Detailed maps and elevation profiles are displayed and accurate based on recorded workout data.

Severity: Major

Priority: High

Type of Testing: Functional Testing, System Testing, Visual Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: FTA-GPS-003

Test Case Title: Verify that notifications are provided for deviations from planned routes.

Test Case Description: Notifications should be provided when a user deviates from a preplanned route.

Test Suite: GPS Functionality

Test Priority: Medium

Preconditions:

- User has created a preplanned route.
- Device has GPS enabled and permission granted for the app.

Test Data: Preplanned route data

Test Steps:

1. Start the fitness tracker app.
2. Enable GPS tracking.
3. Begin following the preplanned route.
4. Deviate from the preplanned route.
5. Verify if a notification is received.

Postconditions:

- Notification is received if deviation from preplanned route is detected.

Expected Result: Notifications are provided for deviations from preplanned routes.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, System Testing, Notification Testing

Test Case Approach: Negative

Test Case 4:

Test Case ID: FTA-GPS-004

Test Case Title: Verify that routes can be saved and revisited.

Test Case Description: Users should be able to save completed routes and revisit them for future workouts.

Test Suite: GPS Functionality

Test Priority: Medium

Preconditions:

- User has completed a workout with GPS tracking enabled.

Test Data: Recorded workout data with GPS tracking enabled

Test Steps:

1. Open the fitness tracker app.
2. Navigate to the "History" or "Completed Workouts" section.
3. Select a workout with GPS tracking enabled.
4. Save the route for future use.
5. Start a new workout and select the saved route.
6. Verify that the app follows the saved route during the workout.

Postconditions:

- Route is saved and can be revisited.

Expected Result: Routes can be saved and revisited, ensuring the app follows the saved route during future workouts.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, System Testing, Regression Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: FTA-GPS-005

Test Case Title: Verify that location-based metrics are included in performance data.

Test Case Description: Location-based metrics, such as speed and elevation, should be included in workout performance data.

Test Suite: GPS Functionality

Test Priority: Low

Preconditions:

- User is outdoors with a clear view of the sky.
- Device has GPS enabled and permission granted for the app.

Test Data: Recorded workout data with GPS tracking enabled

Test Steps:

1. Start the fitness tracker app.
2. Enable GPS tracking.
3. Begin a workout.
4. Complete the workout and save the performance data.
5. Verify that location-based metrics are included.

Postconditions:

- Performance data includes location-based metrics.

Expected Result: Performance data includes accurate and relevant location-based metrics, such as speed, average speed, and elevation gain.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing, System Testing, Performance Testing

Test Case Approach: Positive

Data - 78

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Heart Rate Monitoring

Feature description -

The app monitors heart rate in real-time using built-in sensors or compatible wearable devices, providing insights into exercise intensity, recovery, and cardiovascular health. Heart rate data helps users optimize their workouts, maintain target heart rate zones, and track improvements in fitness over time.

Userstory title -

As a user, I want to monitor my heart rate so that I can optimize my workouts and track my cardiovascular health.

Acceptance criteria -

['Heart rate data is recorded in real-time.', 'Users can set and monitor target heart rate zones.', 'Historical heart rate data is stored and viewable.', 'Alerts are provided for abnormal heart rate readings.', 'Data is compatible with various wearable devices.']

Generated Test Cases:

Test Case ID: FT-HR-001

Test Case Title: Verify that heart rate data is recorded in real-time.

Test Case Description: This test case verifies if the Fitness Tracker App can accurately and consistently record heart rate data in real-time using built-in sensors or compatible wearable devices.

Test Suite: Heart Rate Monitoring

Test Priority: High

Preconditions:

1. User has successfully logged in to the app.

2. Wearable device or built-in sensor is connected and functioning properly.
3. User is at rest or engaged in physical activity.

Test Data: No test data needed

Test Steps:

1. Launch the Fitness Tracker App.
2. Ensure the wearable device or built-in sensor is connected.
3. Start monitoring heart rate on the app.
4. Engage in physical activity, if necessary, to increase heart rate.
5. Observe the heart rate values displayed on the app.

Postconditions:

1. Heart rate values are displayed consistently and accurately.
2. App has not crashed or frozen during heart rate monitoring.

Expected Result: The Fitness Tracker App displays real-time heart rate values consistently and accurately, with no delays or errors.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Real-time Data Testing

Test Case Approach: Positive

Test Case ID: FT-HR-002

Test Case Title: Verify that users can set and monitor target heart rate zones.

Test Case Description: This test case verifies the app's ability for users to set target heart rate zones and monitor their progress in real-time.

Test Suite: Heart Rate Monitoring

Test Priority: High

Preconditions:

1. User has successfully logged in to the app.

2. Wearable device or built-in sensor is connected and functioning properly.
3. User has access to the heart rate zone settings.

Test Data: Target heart rate zones (resting, light, moderate, high, maximum)

Test Steps:

1. Launch the Fitness Tracker App.
2. Ensure the wearable device or built-in sensor is connected.
3. Navigate to heart rate zone settings.
4. Set target heart rate zones for resting, light, moderate, high, and maximum.
5. Start monitoring heart rate on the app.
6. Observe the progress in each target heart rate zone.

Postconditions:

1. App displays real-time progress in each target heart rate zone.
2. User can adjust target heart rate zones as needed.

Expected Result: The Fitness Tracker App allows users to set and monitor target heart rate zones accurately in real-time, with the ability to adjust zones as needed.

Severity: Major

Priority: High

Type of Testing: Functional Testing, User Interface Testing

Test Case Approach: Positive

Test Case ID: FT-HR-003

Test Case Title: Verify that historical heart rate data is stored and viewable.

Test Case Description: This test case checks if the Fitness Tracker App stores historical heart rate data and allows users to view it in a clear and organized manner.

Test Suite: Heart Rate Monitoring

Test Priority: Medium

Preconditions:

1. User has successfully logged in to the app.
2. Wearable device or built-in sensor is connected and functioning properly.
3. User has monitored heart rate data for at least one day.

Test Data: Historical heart rate data

Test Steps:

1. Launch the Fitness Tracker App.
2. Navigate to the heart rate history section.
3. Verify that the app displays heart rate data from the previous day.
4. Observe the data trends (minimum, maximum, average) for the day.
5. Filter the data by week, month, or custom date range.
6. Export the historical heart rate data, if needed.

Postconditions:

1. App displays historical heart rate data accurately.
2. User can filter data by various time ranges.

Expected Result: The Fitness Tracker App stores and displays historical heart rate data accurately, allowing users to filter by time range and export the data if needed.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Data Storage Testing

Test Case Approach: Positive

Test Case ID: FT-HR-004

Test Case Title: Verify that alerts are provided for abnormal heart rate readings.

Test Case Description: This test case verifies if the Fitness Tracker App provides alerts for abnormal heart rate readings, ensuring user safety.

Test Suite: Heart Rate Monitoring

Test Priority: Medium

Preconditions:

1. User has successfully logged in to the app.
2. Wearable device or built-in sensor is connected and functioning properly.
3. App has access to the user's health profile and heart rate norms.

Test Data: Abnormal heart rate values (below resting or above maximum)

Test Steps:

1. Launch the Fitness Tracker App.
2. Ensure the wearable device or built-in sensor is connected.
3. Start monitoring heart rate on the app.
4. Simulate an abnormal heart rate (below resting or above maximum).
5. Observe if the app issues an alert for the abnormal heart rate reading.

Postconditions:

1. App has issued a clear and audible alert for the abnormal heart rate reading.

Expected Result: The Fitness Tracker App provides alerts for abnormal heart rate readings, ensuring user safety and awareness.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Safety Testing

Test Case Approach: Negative

Test Case ID: FT-HR-005

Test Case Title: Verify that data is compatible with various wearable devices.

Test Case Description: This test case checks the Fitness Tracker App's compatibility with various wearable devices, ensuring seamless integration.

Test Suite: Heart Rate Monitoring

Test Priority: High

Preconditions:

1. User has successfully logged in to the app.
2. Multiple compatible wearable devices are available.

Test Data: Compatible wearable devices

Test Steps:

1. Launch the Fitness Tracker App.
2. Connect each available wearable device to the app.
3. Start monitoring heart rate on the app using each connected device.
4. Observe if the app displays accurate and consistent heart rate data for each device.

Postconditions:

1. App displays accurate and consistent heart rate data for each connected device.

Expected Result: The Fitness Tracker App is compatible with various wearable devices, accurately displaying heart rate data for each connected device.

Severity: Major

Priority: High

Type of Testing: Compatibility Testing, Integration Testing

Test Case Approach: Positive

Data - 79

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Calorie Tracking

Feature description -

Users can log their daily food intake and track calorie consumption against calorie expenditure from physical activities, helping them maintain a healthy balance between energy intake and expenditure. This feature promotes mindful eating habits and supports weight management goals.

Userstory title -

As a user, I want to track my calorie intake so that I can maintain a healthy balance between energy intake and expenditure.

Acceptance criteria -

['Users can log daily food intake and calorie consumption.', 'The app calculates and displays total calories consumed and burned.', 'Users can set daily calorie goals and track progress.', 'Food items and calorie information are searchable in the database.', 'Users receive alerts for exceeding calorie goals.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that users can log daily food intake and calorie consumption.

Test Case Description: As a user, I want to be able to add food items and their respective calorie intake into the Fitness Tracker App so that I can monitor my daily calorie consumption.

Test Suite: Calorie Tracking

Test Priority: High

Preconditions:

- User is logged in
- User is on the 'Calorie Tracking' screen

Test Data: Breakfast: oatmeal (200 calories), Lunch: chicken salad (400 calories), Dinner: fish & vegetables (600 calories)

Test Steps:

1. Tap the '+' button to add a new food item.
2. Enter 'Breakfast' as the meal name.
3. Search for 'oatmeal' in the food database and select it.
4. Input 200 as the quantity consumed.
5. Tap 'Save'.
6. Repeat steps 2-5 for Lunch and Dinner.

Postconditions:

- Three food items (Breakfast, Lunch, Dinner) are saved in the user's calorie log.
- Total calorie intake is updated accordingly.

Expected Result: User can successfully log food items and their calorie intake in the calorie log.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that the app calculates and displays total calories consumed and burned.

Test Case Description: As a user, I want the Fitness Tracker App to automatically calculate and display my total calories consumed and burned in a day to help me maintain a healthy energy balance.

Test Suite: Calorie Tracking

Test Priority: High

Preconditions:

- User is logged in
- User has logged food intake and exercise activities

Test Data: Breakfast (200 calories), Lunch (400 calories), Dinner (600 calories), Running (30 minutes - 300 calories)

Test Steps:

1. Navigate to the 'Calorie Tracking' screen.
2. Verify that the app displays the total calories consumed (200 breakfast + 400 lunch + 600 dinner).
3. Verify that the app displays the total calories burned (300 from running).

Postconditions:

- Total calories consumed and burned are displayed accurately.

Expected Result: The Fitness Tracker App calculates and displays the total calories consumed and burned for the user.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users can set daily calorie goals and track progress.

Test Case Description: As a user, I want to set a daily calorie goal to help me manage my weight and maintain a healthy lifestyle. The Fitness Tracker App should allow me to track my progress towards this goal.

Test Suite: Calorie Tracking

Test Priority: Medium

Preconditions:

- User is logged in
- User is on the 'Calorie Tracking' screen

Test Data: Daily calorie goal: 2000 calories

Test Steps:

1. Tap the 'Settings' button on the 'Calorie Tracking' screen.
2. Set the daily calorie goal to 2000 calories.
3. Save settings.
4. Navigate back to the 'Calorie Tracking' screen.

5. Verify that the daily calorie goal is displayed.

6. Log food intake (Breakfast: oatmeal 200 calories, Lunch: chicken salad 400 calories, Dinner: fish & vegetables 600 calories)

7. Verify that the app displays the user's progress towards the daily calorie goal.

Postconditions:

- Daily calorie goal is saved in user settings.
- User's progress towards the daily calorie goal is accurately displayed.

Expected Result: User can set and track a daily calorie goal using the Fitness Tracker App.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that food items and calorie information are searchable in the database.

Test Case Description: As a user, I want to be able to search for food items in the Fitness Tracker App's database and find accurate calorie information so that I can efficiently log my daily food intake.

Test Suite: Calorie Tracking

Test Priority: Medium

Preconditions:

- User is logged in
- User is on the 'Calorie Tracking' screen

Test Data: Search term: 'avocado'

Test Steps:

1. Tap 'Add Food' to add a new food item.
2. Tap the search icon and enter 'avocado' in the search bar.
3. Verify that relevant search results are displayed.

4. Select an avocado item from the search results.

5. Input the quantity consumed and tap 'Save'.

Postconditions:

- Selected food item (avocado) is saved in the user's calorie log.
- Calorie information is accurately displayed for the selected food item.

Expected Result: The Fitness Tracker App's food database is searchable, and users can find accurate calorie information for food items.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that users receive alerts for exceeding calorie goals.

Test Case Description: As a user, I want the Fitness Tracker App to notify me if I exceed my daily calorie goal, so I can make informed decisions about my food intake and physical activity.

Test Suite: Calorie Tracking

Test Priority: Low

Preconditions:

- User is logged in
- User has set a daily calorie goal
- User has logged food intake that exceeds the daily calorie goal

Test Steps:

1. Navigate to the 'Calorie Tracking' screen and verify the daily calorie goal.
2. Log food intake that exceeds the daily calorie goal.
3. Wait for the preconfigured alert time (e.g., 1 hour after exceeding the goal).
4. Verify that a notification or alert is displayed, indicating that the user has exceeded their daily calorie goal.

Postconditions:

- User is notified of exceeding their daily calorie goal.

Expected Result: The Fitness Tracker App notifies users when they exceed their daily calorie goal.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Negative

Data - 80

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Sleep Tracking

Feature description -

The app monitors sleep duration, quality, and patterns using accelerometer and heart rate sensors, providing insights into sleep stages, disturbances, and overall sleep quality. Sleep tracking helps users understand their sleep patterns, identify factors affecting sleep quality, and make lifestyle adjustments for better rest and recovery.

Userstory title -

As a user, I want to monitor my sleep so that I can understand my sleep patterns and improve my sleep

quality.

Acceptance criteria -

['Sleep data is recorded including duration and quality.', 'Users can view detailed sleep stages and patterns.', 'The app provides insights into sleep disturbances.', 'Users receive recommendations for improving sleep quality.', 'Sleep data can be viewed in daily, weekly, and monthly summaries.']

Generated Test Cases:

Test Case ID: FC-ST-001

Test Case Title: Verify that sleep data is recorded including duration and quality.

Test Case Description: This test case verifies the proper recording of sleep data, including duration and quality, in the Fitness Tracker App's sleep tracking feature.

Test Suite: Sleep Tracking

Test Priority: High

Preconditions:

- User has enabled sleep tracking
- User has worn the device or connected a supported wearable for at least one sleep cycle

Test Data: No test data needed

Test Steps:

1. Enable sleep tracking in the app
2. Wear the device or connect a supported wearable device
3. Sleep for at least one cycle
4. Check the sleep data in the app

Postconditions:

- Sleep data is displayed
- Sleep duration and quality are indicated

Expected Result: Sleep data, including duration and quality, are accurately recorded and displayed.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: FC-ST-002

Test Case Title: Verify that users can view detailed sleep stages and patterns.

Test Case Description: This test case checks if sleep stages and patterns are presented accurately within the Fitness Tracker App's sleep tracking feature.

Test Suite: Sleep Tracking

Test Priority: High

Preconditions:

- User has enabled sleep tracking
- User has worn the device or connected a supported wearable for at least one sleep cycle

Test Data: No test data needed

Test Steps:

1. Enable sleep tracking in the app
2. Wear the device or connect a supported wearable device
3. Sleep for at least one cycle
4. View the sleep stages and patterns in the app

Postconditions:

- Sleep stages and patterns are displayed

Expected Result: Sleep stages and patterns are presented accurately without any discrepancies.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: FC-ST-003

Test Case Title: Verify that the app provides insights into sleep disturbances.

Test Case Description: This test case confirms that the Fitness Tracker App's sleep tracking feature correctly detects and shows sleep disturbances.

Test Suite: Sleep Tracking

Test Priority: High

Preconditions:

- User has enabled sleep tracking
- User has worn the device or connected a supported wearable for at least one sleep cycle

Test Data: No test data needed

Test Steps:

1. Enable sleep tracking in the app
2. Wear the device or connect a supported wearable device
3. Sleep for at least one cycle with known disturbances
4. Check sleep disturbance information in the app

Postconditions:

- Sleep disturbances are displayed

Expected Result: Recorded sleep disturbances are displayed accurately in the app.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: FC-ST-004

Test Case Title: Verify that users receive recommendations for improving sleep quality.

Test Case Description: This test case checks if the Fitness Tracker App's sleep tracking feature offers personalized recommendations for users based on their sleep data.

Test Suite: Sleep Tracking

Test Priority: Medium

Preconditions:

- User has enabled sleep tracking
- User has worn the device or connected a supported wearable for at least one week

Test Data: No test data needed

Test Steps:

1. Enable sleep tracking in the app
2. Wear the device or connect a supported wearable device
3. Collect sleep data for one week
4. Check sleep quality recommendations within the app

Postconditions:

- Recommendations are displayed

Expected Result: The app generates tailored sleep quality recommendations based on the user's sleep data.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: FC-ST-005

Test Case Title: Verify that sleep data can be viewed in daily, weekly, and monthly summaries.

Test Case Description: This test case ensures that sleep data can be viewed in different timeframes within the Fitness Tracker App's sleep tracking feature.

Test Suite: Sleep Tracking

Test Priority: Medium

Preconditions:

- User has enabled sleep tracking

- User has worn the device or connected a supported wearable for at least one month

Test Data: No test data needed

Test Steps:

1. Enable sleep tracking in the app
2. Wear the device or connect a supported wearable device
3. Collect sleep data for one month
4. Check sleep data summarized by day, week, and month

Postconditions:

- Sleep data is displayed for each timeframe

Expected Result: Sleep data is displayed in detailed summaries for daily, weekly, and monthly periods.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Regression Testing

Test Case Approach: Positive

Data - 81

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Workout Logging

Feature description -

Users can log individual workouts, specifying exercise type, duration, intensity, and any additional notes or comments. This feature allows users to maintain a detailed workout log, track progress over time, and identify trends in performance and adherence to training programs.

Userstory title -

As a user, I want to log my workouts so that I can track my progress and improve my performance.

Acceptance criteria -

['Users can log various workout types with details.', 'Workout logs include duration, intensity, and notes.', 'Users can edit and delete workout entries.', 'Data is stored and viewable for tracking progress.', 'Users can filter logs by date and workout type.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC001_WorkoutLogging

Test Case Title: Verify that users can log various workout types with details

Test Case Description: As a user, I want to log different workout types with specific details so that I can maintain an accurate and detailed workout log.

Test Suite: Workout Logging

Test Priority: High

Preconditions:

- User is logged in.
- User is on the workout logging page.

Test Data:

- Workout type: Running
- Duration: 30 minutes
- Intensity: Moderate

- Additional notes: Ran in the park.

Test Steps:

1. Navigate to the workout logging page.
2. Select a workout type (e.g., Running).
3. Enter the duration (e.g., 30 minutes).
4. Set the intensity (e.g., Moderate).
5. Add notes (e.g., Ran in the park).
6. Save the workout log.

Postconditions:

- Workout log with the specified information is saved.

Expected Result:

- A new workout log entry is saved, containing the workout type, duration, intensity, and additional notes.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC002_WorkoutLogDurationIntensityNotes

Test Case Title: Verify that workout logs include duration, intensity, and notes

Test Case Description: As a user, I want my workout logs to include duration, intensity, and notes so that I can track and analyze my workouts effectively.

Test Suite: Workout Logging

Test Priority: High

Preconditions:

- User is logged in.

- User has an existing workout log entry.

Test Data: N/A

Test Steps:

1. Navigate to the workout log listing page.
2. Select an existing workout log.
3. Verify the duration, intensity, and notes are displayed.

Postconditions:

- Duration, intensity, and notes are displayed for the selected workout log.

Expected Result:

- The workout log displays the correct duration, intensity, and notes.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC003_EditWorkoutLog

Test Case Title: Verify that users can edit workout entries

Test Case Description: As a user, I want to edit my workout logs to ensure the data is always up-to-date and accurate.

Test Suite: Workout Logging

Test Priority: Medium

Preconditions:

- User is logged in.
- User has an existing workout log entry.

Test Data:

- Updated workout type: Swimming
- Updated duration: 45 minutes
- Updated intensity: High
- Updated notes: Swam in the pool.

Test Steps:

1. Navigate to the workout log listing page.
2. Select an existing workout log.
3. Edit the workout log details (e.g., workout type, duration, intensity, notes).
4. Save the updated workout log.

Postconditions:

- The workout log is updated.

Expected Result:

- The workout log displays the updated details (workout type, duration, intensity, and notes).

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC004_DeleteWorkoutLog

Test Case Title: Verify that users can delete workout entries

Test Case Description: As a user, I want to delete workout logs when they're no longer needed or relevant to ensure my log stays organized.

Test Suite: Workout Logging

Test Priority: Medium

Preconditions:

- User is logged in.
- User has an existing workout log entry.

Test Data: N/A

Test Steps:

1. Navigate to the workout log listing page.
2. Select an existing workout log.
3. Click on the delete button.
4. Confirm the deletion.

Postconditions:

- The workout log is deleted.

Expected Result:

- The workout log no longer appears in the log listing page.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC005_FilterWorkoutLogs

Test Case Title: Verify that users can filter logs by date and workout type

Test Case Description: As a user, I want to filter my workout logs by date and workout type so that I can easily analyze and track specific workouts.

Test Suite: Workout Logging

Test Priority: High

Preconditions:

- User is logged in.

- User has multiple workout log entries with varied workout types and dates.

Test Data:

- Date range: Last week
- Workout type: Running

Test Steps:

1. Navigate to the workout log listing page.
2. Use the filter option to select the desired date range (e.g., Last week).
3. Use the filter option to select the desired workout type (e.g., Running).
4. Click apply filter.

Postconditions:

- Workout logs are filtered by the specified date range and workout type.

Expected Result:

- Only the logs matching the selected date range and workout type are displayed on the log listing page.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 82

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with

wearable devices or fitness equipment.

Feature name -

Goal Setting

Feature description -

Users can set personalized fitness goals based on activity level, weight loss targets, performance milestones, or health objectives. The app provides progress tracking and motivational feedback to help users stay focused and committed to achieving their goals.

Userstory title -

As a user, I want to set fitness goals so that I can stay motivated and track my progress.

Acceptance criteria -

['Users can set goals for various fitness metrics.', 'Progress tracking is available for each goal.', 'Users receive motivational feedback and reminders.', 'Goals can be adjusted or updated as needed.', 'The app provides a summary of goal attainment.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that users can set goals for various fitness metrics

Test Case Description: This test case ensures that users are able to set goals for different fitness metrics such as steps, calories burned, distance traveled, and heart rate.

Test Suite: Goal Setting

Test Priority: High

Preconditions:

- User is logged in
- User has access to the goal setting feature

Test Data: Sample data for each fitness metric

Test Steps:

1. Navigate to the goal setting feature

2. Select a fitness metric (e.g., steps)
3. Enter a specific target value for the selected metric
4. Save and confirm the new goal
5. Repeat steps 2-4 for other fitness metrics

Postconditions:

- Goals are saved and displayed in the user's profile
- New goals are included in the progress tracking feature

Expected Result:

- User is able to set specific targets for different fitness metrics
- Goals are accurately saved and displayed in the user's profile

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that progress tracking is available for each goal

Test Case Description: This test case checks if progress tracking is available for each fitness goal set by a user.

Test Suite: Goal Setting

Test Priority: High

Preconditions:

- User is logged in
- User has set fitness goals in the past

Test Data: Historical user data

Test Steps:

1. Log in as an existing user
2. Navigate to the progress tracking feature

3. Verify that goals and progress are displayed for each fitness metric
4. Check if progress is updated in real-time as fitness activities are logged

Postconditions:

- User can view their progress towards their goals

Expected Result:

- Progress tracking is available for all set goals
- Progress updates are accurate in real-time

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users receive motivational feedback and reminders

Test Case Description: This test case ensures that users receive motivational feedback and reminders to help them achieve their fitness goals.

Test Suite: Goal Setting

Test Priority: Medium

Preconditions:

- User is logged in
- User has set fitness goals

Test Data: None

Test Steps:

1. Log in as an existing user
2. Set fitness goals in the goal setting feature
3. Engage in fitness activities and log them in the app
4. Wait for the app to provide motivational feedback or reminders

Postconditions:

- User receives timely and relevant motivational feedback
- User receives reminders for pending goals or activities

Expected Result:

- User receives motivational feedback and reminders during their fitness journey

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that goals can be adjusted or updated as needed

Test Case Description: This test case tests if users can adjust or update their fitness goals as needed.

Test Suite: Goal Setting

Test Priority: Medium

Preconditions:

- User is logged in
- User has set fitness goals

Test Data: Updated target values for fitness metrics

Test Steps:

1. Log in as an existing user
2. Navigate to the goal setting feature
3. Select a fitness goal to update
4. Change the target value for the selected goal
5. Save and confirm the updated goal

Postconditions:

- Updated goal is saved and displayed in the user's profile

- Progress tracking is updated to reflect the new goal

Expected Result:

- User can adjust or update fitness goals as needed
- Updated goals are accurately reflected in the user's profile and progress tracking

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that the app provides a summary of goal attainment

Test Case Description: This test case verifies if the app provides a summary of goal attainment including completed goals, progress towards incomplete goals, and overall success rate.

Test Suite: Goal Setting

Test Priority: Low

Preconditions:

- User is logged in
- User has set and engaged in fitness activities related to their goals

Test Data: Historical user data

Test Steps:

1. Log in as an existing user
2. Navigate to the goal setting feature
3. Access the summary of goal attainment
4. Review completed goals, progress towards incomplete goals, and overall success rate

Postconditions:

- User can view a summary of their goal attainment

Expected Result:

- App provides an accurate and comprehensive summary of goal attainment

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 83

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Workout Plans

Feature description -

The app offers pre-designed workout plans and training programs tailored to different fitness levels, goals, and preferences. Users can follow structured workout routines, receive guidance on exercise selection and progression, and track their adherence to the plan.

Userstory title -

As a user, I want to follow structured workout plans so that I can achieve my fitness goals with guided training.

Acceptance criteria -

['Users can select workout plans based on fitness levels.', 'Plans include detailed exercise routines and

schedules.', 'Users can track adherence and progress within the plan.', 'The app provides notifications and reminders for workouts.', 'Users can adjust plans based on progress and preferences.']

Generated Test Cases:

Test Case 1:

Test Case ID: FT-WP-001

Test Case Title: Verify that users can select workout plans based on fitness levels.

Test Case Description: This test case verifies that the app allows users to choose a suitable workout plan based on their fitness level.

Test Suite: Workout Plans

Test Priority: High

Preconditions:

- User is registered and logged into the app
- User has access to the Workout Plans feature

Test Data: User profiles with varying fitness levels (Beginner, Intermediate, Advanced)

Test Steps:

1. Log into the Fitness Tracker App as a registered user
2. Navigate to the Workout Plans feature
3. Select a fitness level (Beginner, Intermediate, Advanced)
4. View available workout plans for the selected fitness level

Test Postconditions:

- Workout plans for the selected fitness level are displayed

Expected Result:

- The app displays workout plans tailored to the user's fitness level, with accurate and relevant exercise routines and schedules.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: FT-WP-002

Test Case Title: Verify that plans include detailed exercise routines and schedules.

Test Case Description: This test case verifies that workout plans contain complete and structured exercise routines and schedules.

Test Suite: Workout Plans

Test Priority: High

Preconditions:

- User has access to a workout plan

Test Data: Selected workout plan with a set of exercise routines

Test Steps:

1. Log into the Fitness Tracker App as a registered user
2. Navigate to the Workout Plans feature
3. Select a workout plan
4. Review the exercise routines and schedules for the plan

Test Postconditions:

- The user has reviewed the exercise routines and schedules

Expected Result:

- The selected workout plan contains detailed, structured, and relevant exercise routines and schedules for each day of the plan.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: FT-WP-003

Test Case Title: Verify that users can track adherence and progress within the plan.

Test Case Description: This test case verifies that users can accurately track their adherence and progress while following a workout plan.

Test Suite: Workout Plans

Test Priority: High

Preconditions:

- User is following a workout plan

Test Data: Workout plan activities, completed workouts, and progress metrics

Test Steps:

1. Log into the Fitness Tracker App as a registered user
2. Navigate to the Workout Plans feature
3. Select a workout plan and start following it
4. Complete selected workouts as directed and record activity data
5. Review the progress tracking feature for the workout plan

Test Postconditions:

- User has completed workouts and recorded activity data
- User has reviewed the progress tracking feature

Expected Result:

- The app displays accurate and up-to-date progress metrics for the user's adherence and performance within the workout plan.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: FT-WP-004

Test Case Title: Verify that the app provides notifications and reminders for workouts.

Test Case Description: This test case verifies that the app sends timely notifications and reminders to users to perform workouts.

Test Suite: Workout Plans

Test Priority: Medium

Preconditions:

- User is following a workout plan
- User has configured notification preferences

Test Data: Workout plan schedule

Test Steps:

1. Log into the Fitness Tracker App as a registered user
2. Navigate to the Workout Plans feature
3. Select a workout plan and start following it
4. Ensure notification preferences are set up to receive workout reminders
5. Wait for the scheduled workout time

Test Postconditions:

- User has received a timely notification or reminder

Expected Result:

- The app sends a notification or reminder to the user at the scheduled workout time to encourage adherence to the workout plan.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: FT-WP-005

Test Case Title: Verify that users can adjust plans based on progress and preferences.

Test Case Description: This test case verifies that users can modify their workout plans based on progress and personal preferences.

Test Suite: Workout Plans

Test Priority: High

Preconditions:

- User is following a workout plan
- User has made progress in the plan

Test Data: Workout plan, user progress, and updated plan preferences

Test Steps:

1. Log into the Fitness Tracker App as a registered user
2. Navigate to the Workout Plans feature
3. Select a workout plan and start following it
4. Record progress made in the plan
5. Update workout plan preferences based on progress or personal preferences
6. Review the updated workout plan with the new exercise routines and schedules

Test Postconditions:

- Users have reviewed the updated workout plan with new exercise routines and schedules

Expected Result:

- Users can adjust their workout plans based on progress and preferences, and the app displays the updated exercise routines and schedules accordingly.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Data - 84

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Community Support

Feature description -

Users can connect with a community of like-minded individuals, share progress updates, participate in challenges, and offer support and encouragement to fellow users. Community support fosters accountability, motivation, and a sense of camaraderie among users striving towards similar fitness goals.

Userstory title -

As a user, I want to connect with a community so that I can share my progress and stay motivated.

Acceptance criteria -

['Users can join and participate in community groups.', 'Users can share progress updates and achievements.', 'The app facilitates participation in community challenges.', 'Users can offer and receive support and encouragement.', 'Community activity and engagement are tracked.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that users can join and participate in community groups

Test Case Description: This test case verifies the functionality of joining and participating in community groups within the Fitness Tracker App.

Test Suite: Community Support Module

Test Priority: High

Preconditions:

- User is logged in to the Fitness Tracker App
- User has access to the community groups feature

Test Data: No test data needed

Test Steps:

1. Navigate to the community groups feature in the app.
2. Select a community group to join.
3. Click on the "Join Group" button.
4. Verify that the user is now a member of the selected group.
5. Participate in a discussion or comment on a post within the group.

Postconditions:

- User is a member of the selected community group
- User has successfully participated in a group discussion or commented on a post

Expected Result: User can join and participate in community groups, fostering a sense of community and engagement.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that users can share progress updates and achievements

Test Case Description: This test case verifies the ability of users to share their progress updates and achievements with the community.

Test Suite: Community Support Module

Test Priority: High

Preconditions:

- User is logged in to the Fitness Tracker App
- User has completed a workout or met a goal

Test Data: Pre-prepared workout or goal data

Test Steps:

1. Navigate to the community feed.
2. Click on the "Share Progress" button.
3. Select the completed workout or goal from the list.
4. Add a brief description and optional photo.
5. Post the update to the community feed.
6. Verify that the update is visible to other users in the feed.

Postconditions:

- User's progress update is visible on the community feed.

Expected Result: Users can share their progress updates and achievements, promoting accountability and motivation.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users can participate in community challenges

Test Case Description: This test case verifies the ability of users to participate in community challenges within the Fitness Tracker App.

Test Suite: Community Support Module

Test Priority: High

Preconditions:

- User is logged in to the Fitness Tracker App
- User has access to the community challenges feature

Test Data: No test data needed

Test Steps:

1. Navigate to the community challenges feature in the app.
2. Select an active challenge to join.
3. Review the challenge goals and requirements.
4. Complete the required tasks or workouts to fulfill the challenge.
5. Submit the completed challenge for verification.

Postconditions:

- User's submitted challenge is under review.

Expected Result: Users can join and participate in community challenges, promoting engagement and motivation.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that users can offer and receive support and encouragement

Test Case Description: This test case verifies the ability of users to offer and receive support and encouragement within the Fitness Tracker App.

Test Suite: Community Support Module

Test Priority: High

Preconditions:

- User is logged in to the Fitness Tracker App
- User has access to the community feed

Test Data: Pre-prepared supportive comment

Test Steps:

1. Navigate to the community feed.
2. Locate a user's progress update.
3. Click on the "Comment" button.
4. Enter a supportive and encouraging message.
5. Post the comment.
6. Verify that the comment is visible to the user under their progress update.

Postconditions:

- User's supportive comment is visible on the community feed.

Expected Result: Users can provide and receive support and encouragement through comments on the community feed, fostering a sense of camaraderie and motivation.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that community activity and engagement are tracked

Test Case Description

User

Data - 85

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Integration with Wearable Devices

Feature description -

The app seamlessly integrates with popular wearable devices such as fitness trackers, smartwatches, and heart rate monitors, allowing users to sync activity data and access advanced features. Integration with wearables enhances data accuracy, usability, and convenience for users tracking their fitness activities on the go.

Userstory title -

As a user, I want to integrate my wearable devices so that I can sync my fitness data seamlessly.

Acceptance criteria -

['The app supports integration with multiple wearable devices.', 'Users can sync activity data from their

wearables.', 'The app provides real-time data updates from wearables.', 'Wearable data includes steps, heart rate, and calories burned.', 'Users can troubleshoot connectivity issues with wearables.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC-INT-001

Test Case Title: Verify that the app supports integration with multiple wearable devices

Test Case Description: Verify that the Fitness Tracker App allows users to connect and sync data from multiple wearable devices, such as fitness trackers, smartwatches, and heart rate monitors, ensuring seamless integration and connectivity.

Test Suite: Wearable Device Integration

Test Priority: High

Preconditions:

- User has created an account in the Fitness Tracker App
- User has multiple wearable devices with compatible syncing options

Test Data: Multiple wearable devices with active syncing capabilities

Test Steps:

1. Log into the Fitness Tracker App
2. Navigate to the 'Device Integration' menu
3. Initiate the pairing process for each wearable device
4. Complete the setup for each connected device

Postconditions:

- All devices are connected and visible in the Fitness Tracker App
- Data from all connected devices is accessible within the app

Expected Result: The Fitness Tracker App successfully connects and synchronizes data from multiple wearable devices, providing a seamless integration experience that allows users to access activity data from

their connected devices.

Severity: Major

Priority: High

Type of Testing: Integration Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-INT-002

Test Case Title: Verify that users can sync activity data from their wearables

Test Case Description: Verify that users can manually and automatically sync activity data from their connected wearable devices, ensuring accurate and up-to-date information within the Fitness Tracker App.

Test Suite: Wearable Device Integration

Test Priority: High

Preconditions:

- User has created an account in the Fitness Tracker App
- User has connected at least one wearable device

Test Data: Connected wearable device with active syncing capabilities

Test Steps:

1. Log into the Fitness Tracker App
2. Ensure that at least one wearable device is connected
3. Initiate a manual sync action for the connected device
4. Perform an activity using the connected device for a predetermined amount of time
5. Wait for the automatic sync to occur

Postconditions:

- Manually synced data is visible in the Fitness Tracker App
- Automatically synced data is visible in the Fitness Tracker App

Expected Result: The Fitness Tracker App allows users to manually sync activity data and automatically sync data from connected wearable devices, providing accurate and up-to-date information within the app.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-INT-003

Test Case Title: Verify that the app provides real-time data updates from wearables

Test Case Description: Verify that the Fitness Tracker App receives real-time data updates from connected wearable devices during activities, enabling users to monitor their progress in real time.

Test Suite: Real-Time Wearable Data Updates

Test Priority: High

Preconditions:

- User has created an account in the Fitness Tracker App
- User has connected at least one wearable device

Test Data: Connected wearable device with active syncing capabilities

Test Steps:

1. Log into the Fitness Tracker App
2. Ensure that at least one wearable device is connected
3. Start an activity within the Fitness Tracker App
4. Perform the same activity using the connected wearable device
5. Monitor the Fitness Tracker App for real-time updates from the connected device

Postconditions:

- Real-time data updates are visible in the Fitness Tracker App during the activity

Expected Result: The Fitness Tracker App receives real-time data updates from connected wearable devices during activities, providing users with the ability to monitor their progress in real time within the app.

Severity: Major

Priority: High

Type of Testing: System Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-INT-004

Test Case Title: Verify that wearable data includes steps, heart rate, and calories burned

Test Case Description: Verify that the Fitness Tracker App accurately receives steps, heart rate, and calories burned data from connected wearable devices, ensuring comprehensive activity tracking.

Test Suite: Wearable Data Accuracy

Test Priority: Medium

Preconditions:

- User has created an account in the Fitness Tracker App
- User has connected at least one wearable

Data - 86

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features

like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Water Intake Tracking

Feature description -

Users can monitor their daily water intake and hydration levels, setting hydration goals and receiving reminders to drink water throughout the day. Proper hydration is essential for overall health and performance, and this feature helps users stay hydrated and maintain optimal fluid balance.

Userstory title -

As a user, I want to track my water intake so that I can stay hydrated and maintain optimal health.

Acceptance criteria -

['Users can log their daily water intake.', 'The app calculates total water consumption for the day.', 'Users can set daily hydration goals.', 'Reminders are provided to encourage regular water intake.', 'Users can view hydration progress and trends.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that users can log their daily water intake.

Test Case Description: This test case ensures that users are able to log their daily water intake in the Fitness Tracker App.

Test Suite: Water Intake Tracking

Test Priority: High

Preconditions:

1. User is registered and logged in.
2. User has access to the Water Intake Tracking feature.

Test Data: Water consumption data (e.g., 500ml)

Test Steps:

1. Navigate to the Water Intake Tracking feature.
2. Enter water consumption data in the appropriate field.
3. Save the entered data.

Postconditions:

1. Water intake data is stored in the user's profile.
2. The app calculates total water consumption for the day.

Expected Result: Water intake data is correctly logged, and total water consumption for the day is accurately calculated.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that reminders are provided to encourage regular water intake.

Test Case Description: This test case verifies that the app provides reminders for users to drink water throughout the day.

Test Suite: Water Intake Tracking

Test Priority: Medium

Preconditions:

1. User is registered and logged in.
2. User has set up hydration goals.

Test Data: No test data needed

Test Steps:

1. Navigate to the app's settings.
2. Set up a hydration goal and enable reminders.
3. Wait for the scheduled reminder.

Postconditions:

1. User receives a reminder notification.

Expected Result: User receives a reminder notification at the scheduled time.

Severity: Major

Priority: Medium

Type of Testing: Non-Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users can view hydration progress and trends.

Test Case Description: This test case confirms that users can view their hydration progress and trends over time.

Test Suite: Water Intake Tracking

Test Priority: High

Preconditions:

1. User is registered and logged in.
2. User has logged water intake data for at least a week.

Test Data: No test data needed

Test Steps:

1. Navigate to the Water Intake Tracking feature.
2. Select the 'Progress' or 'Trends' tab.
3. Review hydration data for the specified period.

Postconditions:

1. User can view hydration data for the specified period.

Expected Result: Users can view and analyze their hydration progress and trends over time.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that users can set daily hydration goals.

Test Case Description: This test case verifies that users can set daily hydration goals.

Test Suite: Water Intake Tracking

Test Priority: Medium

Preconditions:

1. User is registered and logged in.
2. User has access to the Water Intake Tracking feature.

Test Data: Hydration goal data (e.g., 2000ml)

Test Steps:

1. Navigate to the Water Intake Tracking feature.
2. Access the app's settings.
3. Set a daily hydration goal.
4. Save the newly set hydration goal.

Postconditions:

1. User's daily hydration goal is saved.

Expected Result: Users can set and save their daily hydration goals.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that the app cannot calculate total water consumption when negative water intake data

is entered.

Test Case Description: This test case confirms that the app handles incorrect water intake data entry (i.e., negative values).

Test Suite: Water Intake Tracking

Test Priority: Low

Preconditions:

1. User is registered and logged in.
2. User has access to the Water Intake Tracking feature.

Test Data: Negative water consumption data (e.g., -500ml)

Test Steps:

1. Navigate to the Water Intake Tracking feature.
2. Enter negative water consumption data in the appropriate field.
3. Save the entered data.

Postconditions:

1. Water intake data is not stored.

Expected Result: The app does not store negative water intake values and does not calculate total water consumption with incorrect data.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Negative

Data - 87

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Nutrition Tracking

Feature description -

The app includes a food diary and nutrition tracker, allowing users to log meals, track macronutrient intake, and monitor nutritional balance. Nutrient tracking provides insight into dietary habits, supports informed food choices, and helps users align their nutrition with their fitness goals.

Userstory title -

As a user, I want to track my nutrition so that I can monitor my diet and make informed food choices.

Acceptance criteria -

['Users can log their daily meals and snacks.', 'The app calculates macronutrient intake (carbs, fats, proteins).', 'Users can set nutritional goals and track progress.', 'The app provides nutritional information and suggestions.', 'Users can view their nutritional history and trends.']

Generated Test Cases:

Test Case ID: TC001

Test Case Title: Verify that users can log their daily meals and snacks.

Test Case Description: This test case verifies that users are able to log their daily meals and snacks within the nutrition tracking feature of the Fitness Tracker App.

Test Suite: Nutrition Tracking

Test Priority: High

Preconditions:

- User is logged into the Fitness Tracker App

- User has access to the nutrition tracking feature

Test Data: Prepare sample meal data including meal name, time consumed, and food items.

Test Steps:

1. Navigate to the nutrition tracking feature.
2. Select the option to log a meal or snack.
3. Enter the meal name, time consumed, and food items.
4. Save the meal entry.
5. Verify that the meal entry is saved and displayed in the nutritional history.

Postconditions:

- Meal entry is saved and displayed correctly
- User can view and edit the meal entry if required

Expected Result: User is able to log meals and snacks, and the entries are saved and displayed correctly in the nutritional history.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that the app calculates macronutrient intake (carbs, fats, proteins).

Test Case Description: This test case verifies that the Fitness Tracker App accurately calculates macronutrient intake (carbs, fats, proteins) for logged meals and snacks.

Test Suite: Nutrition Tracking

Test Priority: High

Preconditions:

- User is logged into the Fitness Tracker App
- User has access to the nutrition tracking feature

- User has logged at least one meal with known macronutrient values

Test Data: Prepare sample meal data with known macronutrient values.

Test Steps:

1. Navigate to the nutrition tracking feature.
2. View the logged meal with known macronutrient values.
3. Verify that the app correctly calculates and displays the macronutrient intake (carbs, fats, proteins) for the meal.

Postconditions:

- Macronutrient intake is calculated correctly and displayed for each meal

Expected Result: The app accurately calculates and displays the macronutrient intake (carbs, fats, proteins) for logged meals and snacks.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users can set nutritional goals and track progress.

Test Case Description: This test case verifies that users can set nutritional goals and track their progress towards these goals within the nutrition tracking feature of the Fitness Tracker App.

Test Suite: Nutrition Tracking

Test Priority: Medium

Preconditions:

- User is logged into the Fitness Tracker App
- User has access to the nutrition tracking feature

Test Data: Prepare sample nutritional goal data.

Test Steps:

1. Navigate to the nutrition tracking feature.
2. Set a nutritional goal for carbs, fats, or proteins.
3. Log meals and snacks to track progress towards the set goal.
4. Verify that the app accurately calculates and displays progress towards the nutritional goal.

Postconditions:

- Nutritional goal is set and displayed correctly
- Progress towards the nutritional goal is calculated and displayed correctly

Expected Result: Users can set nutritional goals and track their progress accurately within the nutrition tracking feature.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that the app provides nutritional information and suggestions.

Test Case Description: This test case verifies that the Fitness Tracker App provides accurate nutritional information and personalized suggestions based on user input.

Test Suite: Nutrition Tracking

Test Priority: Medium

Preconditions:

- User is logged into the Fitness Tracker App
- User has access to the nutrition tracking feature
- User has logged at least one meal with known nutritional values

Test Data: Prepare sample meal data with known nutritional values.

Test Steps:

1. Navigate to the nutrition tracking feature.

2. View the logged meal with known nutritional values.
3. Verify that the app provides accurate nutritional information about the meal.
4. Verify that the app provides personalized suggestions based on the logged meal and user input (e.g., eating habits, nutritional goals).

Postconditions:

- Nutritional information is displayed correctly
- Personalized suggestions are provided based on user input

Expected Result: The app provides accurate nutritional information and personalized suggestions based on user input.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that users can view their nutritional history and trends.

Test Case Description: This test case verifies that users can view their nutritional history and visualize trends in their nutritional intake within the nutrition tracking feature of the Fitness Tracker App.

Test Suite: Nutrition Tracking

Test Priority: Low

Preconditions:

- User is logged into the Fitness Tracker App
- User has access to the nutrition tracking feature
- User has logged at least three meals

Test Steps:

1. Navigate to the nutrition tracking feature.
2. View the nutritional history.

3. Verify that the app displays the summary of macronutrient intake over time.
4. Verify that the app visualizes trends in nutritional intake (e.g., daily, weekly, monthly).

Postconditions:

- Nutritional history is displayed correctly
- Nutritional trends are visualized accurately

Expected Result: Users can view their nutritional history and visualize trends in their nutritional intake.

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 88

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Progress Reports

Feature description -

Users receive regular progress reports and summaries summarizing their activity levels, workout performance, and goal attainment. Progress reports highlight achievements, trends, and areas for improvement, empowering users to make informed decisions and adjustments to their fitness routines.

Userstory title -

As a user, I want to receive progress reports so that I can track my achievements and identify areas for improvement.

Acceptance criteria -

['Users receive weekly and monthly progress reports.', 'Reports include activity levels, workout performance, and goal attainment.', 'Users can view detailed breakdowns of achievements and trends.', 'The app provides recommendations based on report findings.', 'Users can customize the frequency and content of reports.']

Generated Test Cases:

Test Case 1:

Test Case ID: PROGRESS_REPORT_001

Test Case Title: Verify that users receive weekly progress reports.

Test Case Description: This test case verifies that the system generates and delivers weekly progress reports to users as per their preference.

Test Suite: Progress Report Generation

Test Priority: High

Preconditions:

1. User has an account and is logged in.
2. User has set up weekly progress report frequency.

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Access the progress report settings.
3. Set the progress report frequency to weekly.
4. Wait for a week.
5. Check for the presence of a progress report in the user's inbox or notifications.

Postconditions:

1. User receives a weekly progress report.

Expected Result: The user receives a weekly progress report containing activity levels, workout performance, and goal attainment in their inbox or notifications.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Regression Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: PROGRESS_REPORT_002

Test Case Title: Verify that users can view detailed breakdowns of achievements and trends.

Test Case Description: This test case aims to validate that users can access and review detailed breakdowns of their achievements and trends based on the progress reports.

Test Suite: Progress Report Analysis

Test Priority: Medium

Preconditions:

1. User has an account and is logged in.
2. User has generated at least one progress report.

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Access the progress report section.
3. Select a progress report containing achievements and trends.
4. Review the detailed breakdowns of achievements and trends.

Postconditions:

1. User has viewed the detailed breakdowns of achievements and trends.

Expected Result: The user successfully views detailed breakdowns of achievements and trends based on the

selected progress report.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, User Acceptance Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: PROGRESS_REPORT_003

Test Case Title: Verify that the app provides recommendations based on report findings.

Test Case Description: This test case checks whether the system provides personalized recommendations based on users' progress report findings.

Test Suite: Progress Report Recommendations

Test Priority: High

Preconditions:

1. User has an account and is logged in.
2. User has generated at least one progress report.

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Access the progress report section.
3. Select a progress report with areas for improvement.
4. Review the personalized recommendations provided.

Postconditions:

1. User has reviewed the recommendations provided.

Expected Result: The system provides personalized recommendations based on users' progress report findings in the areas for improvement.

Severity: Major

Priority: High

Type of Testing: Functional Testing, User Acceptance Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: PROGRESS_REPORT_004

Test Case Title: Verify that users can customize the frequency and content of reports.

Test Case Description: This test case verifies that users can customize the frequency and content of the progress reports according to their preferences.

Test Suite: Progress Report Customization

Test Priority: Medium

Preconditions:

1. User has an account and is logged in.

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Access the progress report settings.
3. Modify the progress report frequency (e.g., weekly, monthly).
4. Select or deselect desired content categories (e.g., activity levels, workout performance, goal attainment).
5. Save the changes.

Postconditions:

1. User has customized the progress report frequency and content.

Expected Result: The system updates the progress report frequency and content as per the user's customization preferences.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, User Acceptance Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: PROGRESS_REPORT_005

Test Case Title: Verify that users receive monthly progress reports.

Test Case Description: This test case ensures that the system generates and delivers monthly progress reports to users as per their preference.

Test Suite: Progress Report Generation

Test Priority: High

Preconditions:

1. User has an account and is logged in.
2. User has set up monthly progress report frequency.

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Access the progress report settings.
3. Set the progress report frequency to monthly.
4. Wait for a month.
5. Check for the presence of a progress report in the user's inbox or notifications.

Postconditions:

1. User receives a monthly progress report.

Expected Result: The user receives a monthly progress report containing activity levels, workout performance, and

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

In-App Challenges

Feature description -

The app offers in-app challenges and competitions, encouraging users to set personal records, compete with friends, and earn rewards for achieving fitness milestones. In-app challenges add an element of fun, motivation, and friendly competition to the fitness tracking experience.

Userstory title -

As a user, I want to participate in in-app challenges so that I can stay motivated and achieve my fitness goals.

Acceptance criteria -

['Users can join various in-app challenges.', 'Challenges have clear objectives and rules.', 'Users can track their progress within the challenge.', 'Rewards are provided for achieving challenge milestones.', 'Users can compete with friends and view leaderboards.']

Generated Test Cases:

Test Case ID: FTA-IAC-001

Test Case Title: Verify that users can join various in-app challenges

Test Case Description: This test case verifies that users can successfully join different fitness challenges within the app.

Test Suite: User Registration and Challenge Module

Test Priority: High

Preconditions:

- User is logged in to the Fitness Tracker App
- User has access to the "Challenges" section

Test Data:

- User credentials (email and password)
- List of available challenges

Test Steps:

1. Log in to the Fitness Tracker App with valid user credentials.
2. Navigate to the "Challenges" section.
3. Select an available challenge from the list.
4. Click on the "Join Challenge" button.

Postconditions:

- User is added to the selected challenge
- Challenge appears in user's "Active Challenges" list
- User receives a confirmation message or notification

Expected Result:

Upon execution, the user should be able to join any available challenge, and the challenge should appear in the user's list of active challenges.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: FTA-IAC-002

Test Case Title: Verify that challenges have clear objectives and rules

Test Case Description: This test case checks that challenges have clear and well-defined objectives and rules for participants.

Test Suite: Challenge Design and Layout

Test Priority: Medium

Preconditions:

- User is on the challenge details page

Test Data:

- List of challenge details (objectives, rules, duration, etc.)

Test Steps:

1. Navigate to a challenge details page.
2. Review the challenge objectives.
3. Review the challenge rules and requirements.

Postconditions:

- Challenge objectives are clearly stated
- Challenge rules are detailed and easily understandable

Expected Result:

Upon execution, the challenge objectives and rules should be visible, clear, and easily understood by the user.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: FTA-IAC-003

Test Case Title: Verify that users can track their progress within the challenge

Test Case Description: This test case verifies that users can monitor and track their progress in real-time while participating in a challenge.

Test Suite: Progress Tracking and Analytics

Test Priority: High

Preconditions:

- User is logged in to the Fitness Tracker App
- User has joined an active challenge

Test Data:

- User activity data (steps, distance, calories burned, etc.)
- Challenge goals

Test Steps:

1. Log in to the Fitness Tracker App with valid user credentials.
2. Navigate to the active challenge.
3. View the challenge progress.

Postconditions:

- User progress is updated in real-time

- User can view their challenge progress compared to the goal

Expected Result:

Upon execution, users should be able to view their real-time progress in comparison to the challenge goals.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: FTA-IAC-004

Test Case Title: Verify that rewards are provided for achieving challenge milestones

Test Case Description: This test case checks that users receive rewards or recognition upon reaching specific milestones in the challenge.

Test Suite: Rewards and Achievements

Test Priority: High

Preconditions:

- User is logged in to the Fitness Tracker App
- User has joined an active challenge
- User has achieved a challenge milestone

Test Data:

- User activity data
- Challenge milestone requirements

Test Steps:

1. Log in to the Fitness Tracker App with valid user credentials.
2. Navigate to the active challenge.
3. Achieve a milestone in the challenge.
4. View the rewards or achievements section.

Postconditions:

- User receives rewards or recognition for achieving milestones
- Rewards are visible in the user's profile or rewards section

Expected Result:

Upon execution, users should receive rewards or recognition for achieving challenge milestones.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: FTA-IAC-005

Test Case Title: Verify that users can compete with friends and view leaderboards

Test Case Description: This test case verifies that users can view leaderboards and compare their progress with friends.

Test Suite: Friend Integration and Leaderboards

Test Priority: Medium

Preconditions:

- User is logged in to the Fitness Tracker App
- User is connected with friends through the app

- Active challenges include friend competition

Test Data:

- Friend activity data (steps, distance, calories burned, etc.)
- Challenge goals

Test Steps:

1. Log in to the Fitness Tracker App
2. Navigate to an active challenge that includes friend competition
3. View the leaderboard

Postconditions:

- User's position on the leaderboard is accurately represented
- Friends' positions and progress are visible

Expected Result:

Upon execution, users should be able to view leaderboards, see their position, and compare their progress with friends in active challenges.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Customizable Dashboards

Feature description -

Users can customize their app dashboard to display relevant fitness metrics, goals, and progress trackers based on their preferences and priorities. Customizable dashboards provide a personalized user experience, allowing users to focus on the metrics that matter most to them.

Userstory title -

As a user, I want to customize my dashboard so that I can focus on the fitness metrics that are most important to me.

Acceptance criteria -

['Users can select which metrics to display on their dashboard.', 'The app allows customization of dashboard layout and theme.', 'Users can save and update their dashboard settings.', 'Dashboard displays are dynamic and update in real-time.', 'Users can reset the dashboard to default settings.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC-001

Test Case Title: Verify that users can select which metrics to display on their dashboard.

Test Case Description: As a user, I want to customize my dashboard by choosing which fitness metrics to

display.

Test Suite: Customizable Dashboards

Test Priority: High

Preconditions:

- User is logged in.
- User has access to dashboard customization settings.

Test Data: A list of available fitness metrics (e.g., steps taken, calories burned, distance traveled, heart rate)

Test Steps:

1. Log in as a valid user.
2. Navigate to the dashboard customization settings.
3. Select the desired fitness metrics from the list of available options.
4. Save the changes.

Postconditions:

- Selected fitness metrics are displayed on the dashboard.
- Unselected fitness metrics are hidden from the dashboard.

Expected Result: The dashboard displays only the fitness metrics that the user has selected.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-002

Test Case Title: Verify that users can customize the dashboard layout and theme.

Test Case Description: As a user, I want to personalize my dashboard by modifying its layout and appearance.

Test Suite: Customizable Dashboards

Test Priority: High

Preconditions:

- User is logged in.
- User has access to dashboard customization settings.

Test Data: Various dashboard layout templates and theme options

Test Steps:

1. Log in as a valid user.
2. Navigate to the dashboard customization settings.
3. Select a new layout template and apply it.
4. Select a new theme and apply it.
5. Save the changes.

Postconditions:

- New layout and theme have been applied.
- The dashboard displays the selected layout and theme.

Expected Result: The dashboard displays the new layout and theme as chosen by the user.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-003

Test Case Title: Verify that users can save and update their dashboard settings.

Test Case Description: As a user, I want to save and modify my dashboard settings so that I can quickly access my preferred display options.

Test Suite: Customizable Dashboards

Test Priority: Medium

Preconditions:

- User is logged in.
- User has access to dashboard customization settings.

Test Data: Previously saved dashboard settings

Test Steps:

1. Log in as a valid user.
2. Navigate to the dashboard customization settings.
3. Modify the existing dashboard settings.
4. Save the updated settings.

Postconditions:

- Updated dashboard settings are saved.
- The dashboard displays the updated settings.

Expected Result: The dashboard displays the updated settings as saved by the user.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-004

Test Case Title: Verify that dynamic dashboard displays update in real-time.

Test Case Description: As a user, I want my dashboard to display real-time fitness metric updates so that I can monitor my progress effectively.

Test Suite: Customizable Dashboards

Test Priority: High

Preconditions:

- User is logged in.
- User has access to dashboard customization settings.
- Fitness tracking is enabled on wearable devices or fitness equipment.

Test Data: None

Test Steps:

1. Log in as a valid user.
2. Ensure wearable devices or fitness equipment are connected.
3. Perform an activity that updates fitness metrics (e.g., walking, running).

Postconditions:

- The dashboard dynamically updates to display the new fitness metric values.

Expected Result: The dashboard displays the updated fitness metric values in real-time based on the user's activity.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC-005

Test Case Title: Verify that users can reset the dashboard to default settings.

Test Case Description: As a user, I want to reset my dashboard to default settings when I need to start over with a clean slate.

Test Suite: Customizable Dashboards

Test Priority: Low

Preconditions:

- User is logged in.
- User has access to dashboard customization settings.

Test Data: None

Test Steps:

1. Log in as a valid user.
2. Navigate to the dashboard customization settings.
3. Locate and click the "Reset to Default" button.

Postconditions:

- Dashboard settings have been reset to default.
- The dashboard displays default fitness metrics, layout, and theme.

Expected Result: The dashboard displays default fitness metrics, layout, and theme after the user resets it to default settings.

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Negative

Data - 91

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Social Sharing

Feature description -

Users can share workout summaries, achievements, and progress updates on social media platforms, inspiring others and fostering accountability. Social sharing enables users to celebrate their successes, seek support from their social networks, and stay motivated on their fitness journey.

Userstory title -

As a user, I want to share my fitness achievements on social media so that I can inspire others and stay motivated.

Acceptance criteria -

['Users can share workout summaries and achievements.', 'The app supports sharing on multiple social media platforms.', 'Users can customize the content and format of shared posts.', 'Sharing features are integrated seamlessly with social media.', 'Users can track the engagement of their shared posts.']

Generated Test Cases:

Test Case ID: TC-001

Test Case Title: Verify that users can share workout summaries and achievements

Test Case Description: This test case verifies that users can share their workout summaries and fitness achievements on social media platforms through the app.

Test Suite: Social Sharing Module

Test Priority: High

Preconditions:

1. User is logged in.
2. User has completed a workout.
3. User has connected their social media accounts to the app.

Test Data: Pre-existing workout summary and achievement data.

Test Steps:

1. Log in as a user.
2. Complete a workout.
3. Navigate to the workout summary page.
4. Click on the "Share" button.
5. Select the desired social media platform.
6. Customize the content and format of the post, if desired.
7. Confirm the post.

Postconditions:

1. The workout summary and/or achievement should be visible on the selected social media platform.
2. The post engagement should be trackable within the app.

Expected Result: The user can successfully share their workout summaries and achievements on social media platforms.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Social Media Integration Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: TC-002

Test Case Title: Verify that the app supports sharing on multiple social media platforms

Test Case Description: This test case verifies that the app supports and integrates with multiple social media platforms for sharing workout summaries and achievements.

Test Suite: Social Sharing Module

Test Priority: High

Preconditions:

1. User is logged in.

2. User has completed a workout.
3. User has connected their social media accounts to the app.

Test Data: Pre-existing workout summary and achievement data.

Test Steps:

1. Log in as a user.
2. Complete a workout.
3. Navigate to the workout summary page.
4. Click on the "Share" button.
5. Select a different social media platform than the one used in TC-001.
6. Customize the content and format of the post, if desired.
7. Confirm the post.

Postconditions:

1. The workout summary and/or achievement should be visible on the selected social media platform.
2. The post engagement should be trackable within the app.

Expected Result: The user can successfully share their workout summaries and achievements on various social media platforms.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Social Media Integration Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: TC-003

Test Case Title: Verify that users can customize the content and format of shared posts

Test Case Description: This test case verifies that users can customize the content and format of their shared posts on social media platforms.

Test Suite: Social Sharing Module

Test Priority: High

Preconditions:

1. User is logged in.
2. User has completed a workout.
3. User has connected their social media accounts to the app.

Test Data: Pre-existing workout summary and achievement data.

Test Steps:

1. Log in as a user.
2. Complete a workout.
3. Navigate to the workout summary page.
4. Click on the "Share" button.
5. Select a social media platform.
6. Customize the title, description, and image of the post.
7. Confirm the post.

Postconditions:

1. The customized workout summary and/or achievement should be visible on the selected social media platform.

Expected Result: Users can customize the content and format of their shared posts according to their preferences.

Severity: Major

Priority: High

Type of Testing: Functional Testing, User Interface Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: TC-004

Test Case Title: Verify that sharing features are integrated seamlessly with social media

Test Case Description: This test case verifies that the sharing features of the app are integrated smoothly and intuitively with social media platforms.

Test Suite: Social Sharing Module

Test Priority: Medium

Preconditions:

1. User has logged in.
2. User has connected their social media accounts to the app.

Test Data:

No test data needed

Test Steps:

1. Log in to the app and navigate to the workout summary page without completing a workout.
2. Click on the "Share" button.
3. Confirm the social media login if prompted or select another connected account.
4. Notice the warning message or prompt that there is no workout data to share.

Postconditions:

1. The user is redirected back to the workout summary page without sharing any data.

Expected Result: The app's sharing features are seamlessly integrated with social media platforms, including appropriate warnings or prompts when no workout data is present.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, User Interface Testing, Integration Testing, Regression Testing

Test Case Approach: Negative

Test Case ID: TC-005

Test Case Title: Verify that users can track the engagement of their shared posts

Test Case Description: This test case verifies that users can view and track the engagement and interactions of their shared workout summaries and achievements on social media platforms.

Test Suite: Social Sharing Module

Test Priority: Medium

Preconditions:

1. User is logged in.
2. User has shared a workout summary or achievement.

Test Data: Pre-existing shared post data.

Test Steps:

1. Log in as a user.
2. Navigate to the shared posts page.
3. Select a shared post to view its engagement metrics (likes, comments, shares, etc.).

Postconditions:

1. The engagement metrics should be visible and up-to-date.

Expected Result: Users can track and monitor the engagement of their shared workout summaries and achievements on social media platforms using the app.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Social Media Integration Testing, Regression Testing, Data-Driven Testing

Test Case Approach: Positive

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Rest and Recovery Tracking

Feature description -

The app includes features for tracking rest days, recovery activities, and recovery metrics such as resting heart rate and sleep quality. Rest and recovery tracking helps users prioritize recovery, prevent overtraining, and optimize their overall health and performance.

Userstory title -

As a user, I want to track my rest and recovery so that I can optimize my fitness performance and prevent overtraining.

Acceptance criteria -

['Users can log rest days and recovery activities.', 'The app tracks metrics such as resting heart rate and sleep quality.', 'Users receive recommendations for recovery based on their activities.', 'The app provides alerts for insufficient recovery time.', 'Users can view trends and history of their recovery data.']

Generated Test Cases:

Test Case 1:

Test Case ID: FTR-RRT-T001

Test Case Title: Verify that users can log rest days and recovery activities

Test Case Description: Verify the ability for users to log rest days and recovery activities in the Fitness Tracker App.

Test Suite: Rest and Recovery Tracking

Test Priority: High

Preconditions:

- * User is logged in
- * User has access to the rest and recovery tracking feature

Test Data: Rest day entry with relevant notes

Test Steps:

1. Navigate to the rest and recovery tracking feature
2. Select "Log Rest Day" option
3. Enter rest day details and relevant notes
4. Save the rest day entry

Postconditions:

- * Rest day entry is saved and displayed in the user's history

Expected Result:

- * User can successfully log rest days and recovery activities with relevant notes.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: FTR-RRT-T002

Test Case Title: Verify that the app tracks metrics such as resting heart rate and sleep quality

Test Case Description: Verify that the Fitness Tracker App accurately tracks resting heart rate and sleep quality during rest day entries.

Test Suite: Rest and Recovery Tracking

Test Priority: High

Preconditions:

- * User is logged in
- * User has access to the rest and recovery tracking feature
- * User has a compatible wearable device or fitness equipment connected

Test Data: Rest day entry with resting heart rate and sleep quality data

Test Steps:

1. Navigate to the rest and recovery tracking feature
2. Select "Log Rest Day" option
3. Enter resting heart rate and sleep quality data from a connected device
4. Save the rest day entry

Postconditions:

- * Rest day entry is saved with accurate resting heart rate and sleep quality data

Expected Result:

- * The Fitness Tracker App accurately records and displays resting heart rate and sleep quality data from connected devices.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Integration Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: FTR-RRT-T003

Test Case Title: Verify that users receive recommendations for recovery based on their activities

Test Case Description: Verify that the Fitness Tracker App provides personalized recovery recommendations based on users' activities.

Test Suite: Rest and Recovery Tracking

Test Priority: Medium

Preconditions:

- * User is logged in
- * User has access to the rest and recovery tracking feature
- * User has a history of recorded activities

Test Data: User activity history with varying workout intensity and duration

Test Steps:

1. Navigate to the rest and recovery tracking feature
2. View personalized recovery recommendations

Postconditions:

- * Personalized recovery recommendations are displayed based on users' activities

Expected Result:

- * Users receive accurate and relevant recovery recommendations based on their activities and workout intensity.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: FTR-RRT-T004

Test Case Title: Verify that the app provides alerts for insufficient recovery time

Test Case Description: Verify the Fitness Tracker App's ability to alert users when insufficient recovery time has passed.

Test Suite: Rest and Recovery Tracking

Test Priority: High

Preconditions:

- * User is logged in
- * User has access to the rest and recovery tracking feature
- * User has a history of recorded rest day entries

Test Data: User history with insufficient recovery time

Test Steps:

1. Navigate to the rest and recovery tracking feature
2. View alert notifications

Postconditions:

- * Alert notifications are displayed when insufficient recovery time has passed

Expected Result:

- * Users receive accurate and timely alerts for insufficient recovery time.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: FTR-RRT-T005

Test Case Title: Verify that users can view trends and history of their recovery data

Test Case Description: Verify users' ability to view trends and history of their recovery data in the Fitness Tracker App.

Test Suite: Rest and Recovery Tracking

Test Priority: High

Preconditions:

- * User is logged in
- * User has access to the rest and recovery tracking feature
- * User has a history of recorded rest day entries

Test Data: User history with rest day entries

Test Steps:

1. Navigate to the rest and recovery tracking feature
2. View the user's recovery data history
3. Analyze trends in the recovery data

Postconditions:

- * Users can view their recovery data history and analyze trends

Expected Result:

- * Users can effectively view their recovery data history and analyze trends for optimizing their fitness performance and preventing overtraining.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 93

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with

wearable devices or fitness equipment.

Feature name -

Audio Coaching

Feature description -

Users can access audio-guided workouts, coaching cues, and motivational messages during their workouts, enhancing engagement and adherence. Audio coaching provides real-time feedback, encouragement, and guidance, simulating the experience of having a personal trainer or workout buddy.

Userstory title -

As a user, I want to access audio coaching during workouts so that I can stay motivated and receive real-time guidance.

Acceptance criteria -

['Users can access a variety of audio-guided workouts.', 'Audio coaching includes motivational messages and coaching cues.', 'Users can choose coaching based on workout type and intensity.', 'Audio coaching integrates seamlessly with other app features.', 'Users can control audio settings and preferences.']

Generated Test Cases:

Test Case 1:

Test Case ID: FTA-AC-001

Test Case Title: Verify that users can access a variety of audio-guided workouts

Test Case Description: Validate that the app offers multiple audio-guided workout options for users to choose from.

Test Suite: Audio Coaching

Test Priority: High

Preconditions:

- User is logged into the Fitness Tracker App
- User has a stable internet connection

Test Data: No test data needed

Test Steps:

1. Open the Fitness Tracker App
2. Log in using valid credentials
3. Navigate to the Audio Coaching feature
4. Browse available workout options

Expected Result:

- User can access more than five audio-guided workouts across different categories (strength, cardio, yoga, etc.)

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: FTA-AC-002

Test Case Title: Verify that audio coaching includes motivational messages and coaching cues

Test Case Description: Ensure that audio coaching provides motivational messages and coaching cues during workouts.

Test Suite: Audio Coaching

Test Priority: High

Preconditions:

- User is logged into the Fitness Tracker App
- User has a stable internet connection
- User has selected an audio-guided workout

Test Data: No test data needed

Test Steps:

1. Open the Fitness Tracker App

2. Log in using valid credentials
3. Navigate to the Audio Coaching feature
4. Select an audio-guided workout
5. Begin the workout

Expected Result:

- User hears motivational messages and coaching cues at appropriate intervals

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: FTA-AC-003

Test Case Title: Verify that users can choose coaching based on workout type and intensity

Test Case Description: Validate that users can select audio coaching based on their desired workout type and intensity.

Test Suite: Audio Coaching

Test Priority: Medium

Preconditions:

- User is logged into the Fitness Tracker App
- User has a stable internet connection

Test Data: No test data needed

Test Steps:

1. Open the Fitness Tracker App
2. Log in using valid credentials
3. Navigate to the Audio Coaching feature
4. Browse available workout options and filter by workout type and intensity

Expected Result:

- User can select audio coaching based on desired workout type (strength, cardio, yoga, etc.) and intensity (beginner, intermediate, advanced)

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: FTA-AC-004

Test Case Title: Verify that audio coaching integrates seamlessly with other app features

Test Case Description: Ensure that audio coaching integrates well with other features within the Fitness Tracker App.

Test Suite: Audio Coaching

Test Priority: Medium

Preconditions:

- User is logged into the Fitness Tracker App
- User has a stable internet connection
- User has selected an audio-guided workout

Test Data: No test data needed

Test Steps:

1. Open the Fitness Tracker App
2. Log in using valid credentials
3. Navigate to the Audio Coaching feature
4. Select an audio-guided workout
5. Begin the workout
6. Use other app features (heart rate monitor, workout log, etc.) during the workout

Expected Result:

- Audio coaching continues playing without interruption while using other app features

Severity: Major

Priority: Medium

Type of Testing: Integration Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: FTA-AC-005

Test Case Title: Verify that users can control audio settings and preferences

Test Case Description: Validate that users can adjust audio settings and preferences as needed.

Test Suite: Audio Coaching

Test Priority: Low

Preconditions:

- User is logged into the Fitness Tracker App
- User has a stable internet connection

Test Data: No test data needed

Test Steps:

1. Open the Fitness Tracker App
2. Log in using valid credentials
3. Navigate to the Audio Coaching feature
4. Begin playing an audio-guided workout
5. Access the audio settings menu and adjust settings (volume, playback speed, etc.)

Expected Result:

- User can adjust audio settings and preferences as desired

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 94

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Integration with Health Apps

Feature description -

The app integrates with health and wellness apps such as calorie counters, meditation apps, and mental health trackers, enabling users to consolidate their health data and gain a comprehensive view of their overall well-being. Integration with health apps promotes holistic health management and synergy between different aspects of wellness.

Userstory title -

As a user, I want to integrate the fitness tracker app with other health apps so that I can have a comprehensive view of my well-being.

Acceptance criteria -

['The app integrates with popular health and wellness apps.', 'Users can sync data from connected health apps.', 'The app provides a consolidated view of health metrics.', 'Users receive insights based on combined data from multiple apps.', 'Integration settings are easy to manage and configure.']

Generated Test Cases:

Test Case ID: TC-101

Test Case Title: Verify that the fitness tracker app integrates with popular health and wellness apps

Test Case Description: This test case verifies that the fitness tracker app can connect and exchange data with popular health and wellness apps.

Test Suite: Integration Testing

Test Priority: High

Preconditions:

- Internet connection is available
- User is logged in to the fitness tracker app
- User has permissions to install and use other health apps

Test Data: List of popular health and wellness apps

Test Steps:

1. Install and open the fitness tracker app.
2. Log in using valid credentials.
3. Access the app's integration settings.
4. Search for and select a popular health app from the list.
5. Follow the prompts to authorize data sharing between the two apps.
6. Verify that the selected health app appears in the list of connected apps within the fitness tracker app.

Postconditions:

- Selected health app is connected and data sharing is enabled
- Fitness tracker app displays a confirmation message indicating a successful connection

Expected Result: The fitness tracker app integrates successfully with at least one popular health and wellness app.

Severity: Major

Priority: High

Type of Testing: Integration Testing

Test Case Approach: Positive

Test Case ID: TC-102

Test Case Title: Verify that users can sync data from connected health apps

Test Case Description: This test case confirms that users can import and export data from connected health apps within the fitness tracker app.

Test Suite: Integration Testing

Test Priority: High

Preconditions:

- Internet connection is available
- User is logged in to the fitness tracker app
- At least one health app is connected and sharing data with the fitness tracker app

Test Data: Health data from connected apps

Test Steps:

1. Open the fitness tracker app and navigate to the data sync section.
2. Select a connected health app from the list.
3. Initiate a data sync between the two apps.
4. Monitor the progress of the data sync.
5. Verify that health data from the selected app is displayed within the fitness tracker app.

Postconditions:

- Data from the connected health app is successfully imported and displayed
- Fitness tracker app confirms a successful data sync

Expected Result: Users can sync data from connected health apps within the fitness tracker app.

Severity: Major

Priority: High

Type of Testing: Integration Testing

Test Case Approach: Positive

Test Case ID: TC-103

Test Case Title: Verify that the app provides a consolidated view of health metrics

Test Case Description: This test case checks that the fitness tracker app aggregates and displays health metrics from connected apps.

Test Suite: Functional Testing

Test Priority: High

Preconditions:

- Internet connection is available
- User is logged in to the fitness tracker app
- At least two health apps are connected and sharing data with the fitness tracker app

Test Data: Health data from connected apps

Test Steps:

1. Open the fitness tracker app and navigate to the dashboard.
2. Verify that health metrics such as steps, calories burned, distance traveled, and heart rate are displayed.
3. Verify that these health metrics originate from both connected health apps.
4. Verify that the aggregated health metrics are accurate and up-to-date.

Postconditions:

- Health metrics from connected apps are displayed in a consolidated format
- Fitness tracker app confirms successful aggregation of data from multiple apps

Expected Result: The fitness tracker app provides a consolidated view of health metrics from connected apps.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-104

Test Case Title: Verify that users receive insights based on combined data from multiple apps

Test Case Description: This test case verifies that users receive personalized insights based on consolidated health data from multiple apps.

Test Suite: Functional Testing

Test Priority: Medium

Preconditions:

- Internet connection is available
- User is logged in to the fitness tracker app
- At least two health apps are connected and sharing data with the fitness tracker app
- User has provided sufficient health data for analysis

Test Data: Health data from connected apps

Test Steps:

1. Open the fitness tracker app and navigate to the insights section.
2. Verify that personalized insights based on health data from multiple apps are displayed.
3. Analyze the insights and verify they are accurate and relevant.

Postconditions:

- Personalized insights are generated using health data from connected apps
- Fitness tracker app confirms successful generation of insights

Expected Result: Users receive insights based on combined data from multiple apps.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-105

Test Case Title: Verify that integration settings are easy to manage and configure

Test Case Description: This test case ensures that users can easily manage and configure integration settings for connecting and disconnecting health apps.

Test Suite: User Interface Testing

Test Priority: Low

Preconditions:

- User is logged in to the fitness tracker app

Test Data: No test data needed

Test Steps:

1. Open the fitness tracker app and navigate to the integration settings.
2. Verify that the process for connecting a health app is intuitive.
3. Verify that the process for disconnecting a health app is also intuitive.
4. Verify that the app displays a clear confirmation message upon successful connection or disconnection.
5. Verify that the health app status (connected or disconnected) is accurately reflected in the app.

Postconditions:

- User has successfully managed and configured integration settings
- Fitness tracker app confirms successful management and configuration of integration settings

Expected Result: Integration settings are easy to manage and configure.

Severity: Minor

Priority: Low

Type of Testing: User Interface Testing

Test Case Approach: Positive

Data - 95

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Body Composition Tracking

Feature description -

Users can track changes in body composition metrics such as weight, body fat percentage, muscle mass, and BMI over time. Body composition tracking provides insights into changes in physique, fitness progress, and the effectiveness of exercise and nutrition interventions.

Userstory title -

As a user, I want to track my body composition so that I can monitor my fitness progress and make informed decisions about my training and nutrition.

Acceptance criteria -

['Users can log body composition metrics such as weight, body fat percentage, and muscle mass.', 'The app calculates and displays BMI based on logged data.', 'Users can view trends and history of body composition changes.', 'The app provides insights and recommendations based on body composition data.', 'Users can set goals for body composition metrics.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC-BC-001

Test Case Title: Verify that users can log body composition metrics such as weight, body fat percentage, and muscle mass.

Test Case Description: This test case verifies that the user is able to input and save body composition data,

including weight, body fat percentage, and muscle mass.

Test Suite: Body Composition Tracking

Test Priority: High

Preconditions: User is logged in, user is on the body composition tracking page

Test Data: Weight: 160 lbs, Body Fat Percentage: 15%, Muscle Mass: 120 lbs

Test Steps:

1. Navigate to the body composition tracking page
2. Enter the weight as 160 lbs
3. Enter the body fat percentage as 15%
4. Enter the muscle mass as 120 lbs
5. Click the "Save" button

Postconditions: Body composition data (weight, body fat percentage, muscle mass) is saved and displayed on the body composition tracking page

Expected Result: Body composition data is saved and displayed on the body composition tracking page

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-BC-002

Test Case Title: Verify that the app calculates and displays BMI based on logged data.

Test Case Description: This test case verifies that the app calculates and displays the user's BMI based on their logged weight and height.

Test Suite: Body Composition Tracking

Test Priority: High

Preconditions: User is logged in, user has logged weight and height data

Test Data: Weight: 160 lbs, Height: 5'10"

Test Steps:

1. Navigate to the body composition tracking page
2. Verify that the user's height is displayed
3. Verify that the user's weight is displayed
4. Calculate the user's BMI manually
5. Compare the calculated BMI to the BMI displayed in the app

Postconditions: The user's BMI is accurately calculated and displayed

Expected Result: The user's BMI is accurately calculated and displayed

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-BC-003

Test Case Title: Verify that users can view trends and history of body composition changes.

Test Case Description: This test case verifies that the user is able to view their body composition data over time, including changes in weight, body fat percentage, and muscle mass.

Test Suite: Body Composition Tracking

Test Priority: Medium

Preconditions: User is logged in, user has logged at least two sets of body composition data

Test Data: N/A

Test Steps:

1. Navigate to the body composition tracking page

2. View the graph displaying body composition data over time
3. Verify that the data points on the graph correspond to the user's logged data
4. Verify that the user is able to zoom in and out on the graph

Postconditions: The user is able to view their body composition data over time

Expected Result: The user is able to view their body composition data over time

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-BC-004

Test Case Title: Verify that the app provides insights and recommendations based on body composition data.

Test Case Description: This test case verifies that the app provides the user with insights and recommendations based on their body composition data.

Test Suite: Body Composition Tracking

Test Priority: Low

Preconditions: User is logged in, user has logged at least two sets of body composition data

Test Data: N/A

Test Steps:

1. Navigate to the body composition tracking page
2. View the insights and recommendations displayed on the page
3. Verify that the insights and recommendations are relevant to the user's body composition data
4. Verify that the insights and recommendations change as the user's body composition data changes

Postconditions: The user is provided with relevant insights and recommendations based on their body composition data

Expected Result: The user is provided with relevant insights and recommendations based on their body composition data

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC-BC-005

Test Case Title: Verify that users can set goals for body composition metrics.

Test Case Description: This test case verifies that the user is able to set goals for their body composition metrics, including weight, body fat percentage, and muscle mass.

Test Suite: Body Composition Tracking

Test Priority: Medium

Preconditions: User is logged in

Test Data: Weight goal: 155 lbs, Body Fat Percentage goal: 12%, Muscle Mass goal: 125 lbs

Test Steps:

1. Navig

Data - 96

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health

metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Workout Analysis

Feature description -

The app offers detailed workout analysis tools, including performance metrics, trends, and insights into exercise intensity, duration, and calorie burn. Workout analysis helps users assess the effectiveness of their workouts, identify areas for improvement, and optimize training strategies for better results.

Userstory title -

As a user, I want to analyze my workouts so that I can assess my performance and optimize my training strategies.

Acceptance criteria -

['Users can view detailed performance metrics for each workout.', 'The app provides trends and insights into workout intensity and duration.', 'Users can track calorie burn and other health metrics.', 'The app offers recommendations based on workout analysis.', 'Users can compare current workouts with past performance.']

Generated Test Cases:

Test Case 1:

Test Case ID: FTA-WKA-001

Test Case Title: Verify that users can view detailed performance metrics for each workout

Test Case Description: This test case verifies that the fitness tracker app displays accurate and detailed workout performance metrics for individual workouts.

Test Suite: Workout Analysis

Test Priority: High

Preconditions:

- User has created a workout session and saved it.

- User has navigated to the workout session details page.

Test Data: Previously saved workout session

Test Steps:

1. Open the fitness tracker app.
2. Navigate to the list of completed workout sessions.
3. Select a specific workout session.
4. Observe the performance metrics, such as calories burned, distance traveled, and heart rate.

Postconditions:

- Workout performance metrics are displayed accurately.

Expected Result:

- Workout performance metrics are displayed correctly and in a user-friendly format.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: FTA-WKA-002

Test Case Title: Verify that the app provides trends and insights into workout intensity and duration

Test Case Description: This test case confirms if the fitness tracker app generates trends and provides insights into the user's workout intensity and duration.

Test Suite: Workout Analysis

Test Priority: High

Preconditions:

- User has completed multiple workout sessions of different types and intensities.

Test Data: Multiple workout sessions

Test Steps:

1. Open the fitness tracker app.
2. Navigate to the workout analysis dashboard.
3. Check the provided trends and insights based on workout intensity and duration.

Postconditions:

- Trends and insights are generated and displayed on the dashboard.

Expected Result:

- The app displays accurate trends and insights about workout intensity and duration.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: FTA-WKA-003

Test Case Title: Verify that users can track calorie burn and other health metrics

Test Case Description: This test case ensures the fitness tracker app tracks and displays calorie burn and other health metrics accurately.

Test Suite: Workout Analysis

Test Priority: High

Preconditions:

- User has completed a workout session.

Test Data: Previously saved workout session

Test Steps:

1. Open the fitness tracker app.
2. Navigate to the completed workout session.
3. Observe calorie burn and other health metrics, such as steps and heart rate.

Postconditions:

- Calorie burn and health metrics are displayed correctly.

Expected Result:

- The app displays accurate calorie burn and health metrics.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: FTA-WKA-004

Test Case Title: Verify that the app offers recommendations based on workout analysis

Test Case Description: This test case verifies if the fitness tracker app provides personalized recommendations based on the user's workout analysis.

Test Suite: Workout Analysis

Test Priority: High

Preconditions:

- User has completed multiple workout sessions.

Test Data: Multiple workout sessions

Test Steps:

1. Open the fitness tracker app.
2. Navigate to the workout analysis dashboard.
3. Check if personalized workout recommendations are provided based on the trends and insights.

Postconditions:

- Personalized workout recommendations are displayed based on trends and insights.

Expected Result:

- The app provides accurate and relevant workout recommendations based on user's workout analysis.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: FTA-WKA-005

Test Case Title: Verify that users can compare current workouts with past performance

Test Case Description: This test case checks if the fitness tracker app allows users to compare their current workout performance with past records.

Test Suite: Workout Analysis

Test Priority: High

Preconditions:

- User has completed multiple workout sessions.

Test Data: Multiple workout sessions

Test Steps:

1. Open the fitness tracker app.
2. Navigate to the list of completed workout sessions.
3. Select a recent workout session.
4. Compare the performance metrics with previous workout sessions.

Postconditions:

- Current workout performance can be compared with past sessions.

Expected Result:

- The app enables users to compare current and past workout performance data.

Severity: Major

Priority: High

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Weather Integration

Feature description -

Integrated weather forecasts and conditions allow users to plan outdoor workouts and activities based on current weather conditions and forecasts. Weather integration helps users make informed decisions, stay safe, and optimize their outdoor exercise routines according to weather conditions.

Userstory title -

As a user, I want to view weather forecasts in the app so that I can plan my outdoor workouts accordingly.

Acceptance criteria -

['The app provides current weather conditions and forecasts.', 'Users can view weather details for different locations.', 'Weather data is updated in real-time.', 'The app offers recommendations based on weather conditions.', 'Users can receive weather alerts for extreme conditions.']

Generated Test Cases:

Test Case 1:

Test Case ID: FTA-WT-001

Test Case Title: Verify that the app provides current weather conditions and forecasts.

Test Case Description: This test case verifies that the fitness tracker app displays accurate and up-to-date

weather information, including current weather conditions and forecasts, for the user's location.

Test Suite: Weather Integration

Test Priority: High

Preconditions:

- User is logged in.
- User has allowed location access.

Test Data: No test data needed

Test Steps:

1. Open the Fitness Tracker App.
2. Navigate to the Weather Integration feature.
3. Allow location access if prompted.

Expected Result: The app displays accurate current weather conditions and a reliable forecast for the user's location.

Severity: Major

Priority: High

Type of Testing: Functional Testing, System Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: FTA-WT-002

Test Case Title: Verify that users can view weather details for different locations.

Test Case Description: This test case checks if users can view weather details for any location of their choice, allowing them to plan workouts in other areas.

Test Suite: Weather Integration

Test Priority: Medium

Preconditions:

- User is logged in.

- User has internet access.

Test Data: A specific address or postal code.

Test Steps:

1. Open the Fitness Tracker App.
2. Navigate to the Weather Integration feature.
3. Enter a different location (address or postal code).

Expected Result: The app displays accurate weather conditions and forecasts for the entered location.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, System Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: FTA-WT-003

Test Case Title: Verify that weather data is updated in real-time.

Test Case Description: This test case verifies the Fitness Tracker App updates weather data in real-time, ensuring users have access to the latest weather information.

Test Suite: Weather Integration

Test Priority: High

Preconditions:

- User is logged in.
- User has allowed location access.

Test Data: No test data needed

Test Steps:

1. Open the Fitness Tracker App.
2. Navigate to the Weather Integration feature.
3. Allow location access if prompted.

4. Observe the current weather conditions displayed.

5. Wait for approximately 5-10 minutes.

Expected Result: The app updates the current weather conditions to match real-time information.

Severity: Critical

Priority: High

Type of Testing: Functional Testing, System Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: FTA-WT-004

Test Case Title: Verify that the app offers recommendations based on weather conditions.

Test Case Description: This test case verifies the Fitness Tracker App offers workout recommendations based on current weather conditions to ensure user safety and workout effectiveness.

Test Suite: Weather Integration

Test Priority: Medium

Preconditions:

- User is logged in.
- User has allowed location access.

Test Data: No test data needed

Test Steps:

1. Open the Fitness Tracker App.
2. Navigate to the Weather Integration feature.
3. Allow location access if prompted.

Expected Result: The app displays workout recommendations tailored to the current weather conditions.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, System Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: FTA-WT-005

Test Case Title: Verify that users can receive weather alerts for extreme conditions.

Test Case Description: This test case verifies the Fitness Tracker App notifies users of extreme weather alerts, allowing them to adjust their workout plans accordingly.

Test Suite: Weather Integration

Test Priority: Medium

Preconditions:

- User is logged in.
- User has allowed location access and notification permissions.

Test Data: No test data needed

Test Steps:

1. Open the Fitness Tracker App.
2. Navigate to the Weather Integration feature.
3. Allow location access and notification permissions if prompted.
4. Simulate extreme weather conditions.

Expected Result: The app sends a weather alert notification to the user, warning them of the extreme conditions.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, System Testing

Test Case Approach: Negative

Project name -

Fitness Tracker App

Project description -

Create a fitness tracker app that allows users to track their physical activity, exercise routines, and health metrics such as steps taken, calories burned, distance traveled, and heart rate. The app should include features like workout logging, goal setting, progress tracking, personalized recommendations, and integration with wearable devices or fitness equipment.

Feature name -

Mood Tracking

Feature description -

Users can track their mood, energy levels, and motivation before and after workouts, providing context for their exercise experiences and insights into the impact of physical activity on mental well-being. Mood tracking promotes self-awareness, mindfulness, and emotional regulation in the context of fitness and exercise.

Userstory title -

As a user, I want to track my mood and energy levels before and after workouts so that I can understand the impact of exercise on my mental well-being.

Acceptance criteria -

['Users can log their mood, energy levels, and motivation.', 'The app provides insights into mood and energy trends over time.', 'Users receive recommendations based on their mood and energy data.', 'The app supports journaling for personal reflections on mood and workouts.', 'Users can set goals for improving mood and energy levels.']

Generated Test Cases:

Test Case 1:

Test Case ID: FTA-MT-001

Test Case Title: Verify that users can log their mood, energy levels, and motivation.

Test Suite: Mood Tracking

Test Priority: High

Preconditions: User is logged in, user has access to the mood tracking feature.

Test Data: Predefined mood, energy levels, and motivation values.

Test Steps:

1. Navigate to the mood tracking feature.
2. Select a mood value from the list of options.
3. Select an energy level value from the list of options.
4. Select a motivation level value from the list of options.
5. Save the entered values.

Postconditions: Entered mood, energy levels, and motivation values are saved and displayed in the user's history.

Expected Result: User is able to log their mood, energy levels, and motivation successfully.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: FTA-MT-002

Test Case Title: Verify that the app provides insights into mood and energy trends over time.

Test Suite: Mood Tracking

Test Priority: Medium

Preconditions: User has logged their mood and energy levels for at least one week.

Test Data: Previously saved mood and energy levels data.

Test Steps:

1. Navigate to the mood tracking feature.
2. Access the insights or analytics section.

3. Observe the visual representation of mood and energy levels.

Postconditions: The app displays a visual representation of the user's mood and energy levels over time.

Expected Result: The app provides insights and trends on the user's mood and energy levels.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: FTA-MT-003

Test Case Title: Verify that users receive recommendations based on their mood and energy data.

Test Suite: Mood Tracking

Test Priority: Medium

Preconditions: User has logged their mood, energy levels, and motivation for at least one week.

Test Data: Previously saved mood, energy levels, and motivation data.

Test Steps:

1. Navigate to the mood tracking feature.
2. Access the recommendations or tips section.
3. Observe personalized recommendations.

Postconditions: App displays personalized recommendations based on the user's mood and energy data.

Expected Result: The app provides recommendations based on the user's historical mood and energy levels.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: FTA-MT-004

Test Case Title: Verify that the app supports journaling for personal reflections on mood and workouts.

Test Suite: Mood Tracking

Test Priority: Low

Preconditions: User is logged in, user has access to the journaling feature.

Test Data: Predefined journal entries.

Test Steps:

1. Navigate to the journaling feature.
2. Enter a journal entry related to the user's mood or workout.
3. Save the journal entry.

Postconditions: Journal entry is saved and accessible in the user's history.

Expected Result: The app supports journaling for the user's personal reflections on mood and workouts.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: FTA-MT-005

Test Case Title: Verify that users can set goals for improving mood and energy levels.

Test Suite: Mood Tracking

Test Priority: Medium

Preconditions: User is logged in, user has access to the goal setting feature.

Test Data: Predefined mood or energy level improvement goals.

Test Steps:

1. Navigate to the goal setting feature.
2. Select either mood or energy levels as the area for improvement.

3. Set a specific goal and a target date.

4. Save the goal.

Postconditions: Newly created goal is displayed in the user's list of goals.

Expected Result: Users can set and track goals for improving their mood and energy levels.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 99

Project name -

Recipe Sharing Platform

Project description -

Build a recipe sharing platform where users can discover, save, and share recipes, cooking tips, and culinary inspiration. The platform should include features like recipe search and browse, user-generated content, ratings and reviews, meal planning, grocery list generation, and social sharing. Users should be able to upload their own recipes, photos, and videos, and engage with other users' content through comments and likes.

Feature name -

Recipe Creation

Feature description -

Users can create and share their own recipes on the platform, including ingredients, cooking instructions, and optional photos. This feature allows users to showcase their culinary creations and contribute to the platform's recipe database.

Userstory title -

As a user, I want to create and share my own recipes so that I can showcase my culinary creations.

Acceptance criteria -

['Users can input and save ingredients and cooking instructions.', 'Users can upload photos for their recipes.',
"Recipes are saved in the user's profile.", 'Recipes are accessible to other users on the platform.', 'Users receive
a confirmation upon successful recipe submission.']

Generated Test Cases:

Test Case 1:

Test Case ID: TC-REC001

Test Case Title: Verify that users can input and save ingredients for a recipe

Test Case Description: This test case will validate if users can input and save ingredients while creating a new recipe.

Test Suite: Recipe Creation

Test Priority: High

Preconditions:

- User is logged in
- User is on the recipe creation page

Test Data:

- Recipe name
- List of ingredients
- Cooking instructions

Test Steps:

1. Navigate to the recipe creation page.
2. Input a recipe name.
3. Enter the list of ingredients.
4. Save the recipe.

Postconditions:

- Recipe is saved in the user's profile.
- Ingredients are saved with the recipe.

Expected Result:

- Recipe is saved successfully, and ingredients are displayed correctly within the user's profile and on the recipe page.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-REC002

Test Case Title: Verify that users can upload photos for their recipes

Test Case Description: This test case validates that users can upload a photo while creating a new recipe.

Test Suite: Recipe Creation

Test Priority: High

Preconditions:

- User is logged in
- User is on the recipe creation page

Test Data:

- Recipe name

- Ingredients
- Cooking instructions
- Photo for the recipe

Test Steps:

1. Navigate to the recipe creation page.
2. Input a recipe name.
3. Enter the ingredients and cooking instructions.
4. Upload a photo for the recipe.
5. Save the recipe.

Postconditions:

- Recipe is saved in the user's profile.
- Photo is displayed with the recipe.

Expected Result:

- Recipe is saved successfully, and the uploaded photo is displayed correctly within the user's profile and on the recipe page.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-REC003

Test Case Title: Verify that recipes are saved in the user's profile

Test Case Description: This test case will confirm that recipes are saved correctly in the user's profile upon submission.

Test Suite: Recipe Creation

Test Priority: High

Preconditions:

- User is logged in
- User has created a recipe with ingredients and cooking instructions

Test Steps:

1. Navigate to the user's profile page.
2. Verify that the new recipe displays in the user's list of recipes.

Postconditions:

- Recipe is visible in the user's profile.

Expected Result:

- The newly created recipe is displayed in the user's profile list, including all input ingredients and cooking instructions.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-REC004

Test Case Title: Verify that recipes are accessible to other users on the platform

Test Case Description: This test case ensures that recipes created by a user are accessible and visible to other users on the platform.

Test Suite: Recipe Sharing

Test Priority: Medium

Preconditions:

- User A has created a recipe
- User B is logged in

Test Steps:

1. Navigate to User A's profile as User B.
2. Verify that User A's recipe is accessible and visible to User B.

Postconditions:

- Recipe is visible and accessible to other users.

Expected Result:

- User B can view and access User A's recipe, including ingredients and cooking instructions.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC-REC005

Test Case Title: Verify that users receive a confirmation upon successful recipe submission

Test Case Description: This test case validates that users receive a confirmation message after submitting a new recipe.

Test Suite: Recipe Creation

Test Priority: High

Preconditions:

- User is logged in
- User is on the recipe creation page
- User has input a recipe name, ingredients, and cooking instructions

Test Steps:

1. Save the new recipe by clicking the submit button.
2. Verify that a confirmation message indicating successful recipe submission appears.

Postconditions:

- Confirmation message is displayed.
- Recipe is saved in the user's profile.

Expected Result:

- User sees a confirmation message indicating that the recipe was successfully submitted and saved in the user's profile.

Severity: Minor

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Data - 100

Project name -

Recipe Sharing Platform

Project description -

Build a recipe sharing platform where users can discover, save, and share recipes, cooking tips, and culinary inspiration. The platform should include features like recipe search and browse, user-generated content, ratings and reviews, meal planning, grocery list generation, and social sharing. Users should be able to upload their own recipes, photos, and videos, and engage with other users' content through comments and likes.

Feature name -

Recipe Discovery

Feature description -

The platform offers a robust search functionality that allows users to discover new recipes based on keywords, ingredients, cuisine types, dietary preferences, and more. Users can explore a wide variety of recipes to find inspiration for their next meal.

Userstory title -

As a user, I want to search for recipes using keywords and ingredients so that I can find new recipes to try.

Acceptance criteria -

['Users can search for recipes using keywords and ingredients.', 'Search results are relevant and accurately match the search criteria.', 'Users can filter search results by cuisine type, dietary preferences, and other criteria.', 'Recipes in the search results are accessible and detailed.', 'Users can save search results for future reference.']

Generated Test Cases:

Test Case ID: TC-RD-001

Test Case Title: Verify that users can search for recipes using keywords

Test Case Description: This test case verifies that the recipe sharing platform allows users to search for recipes using keywords.

Test Suite: Recipe Discovery

Test Priority: High

Preconditions:

- User is logged in or on the public search page
- Recipes with various keywords have been added to the system

Test Data: Several recipe keywords (e.g. pasta, chicken, vegetarian)

Test Steps:

1. Navigate to the search bar
2. Enter a recipe keyword (e.g. pasta) in the search bar
3. Click the 'Search' button or press Enter

Postconditions:

- User is presented with a list of recipes related to the entered keyword
- The search bar maintains the entered keyword

Expected Result: Recipes related to the entered keyword are displayed

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-RD-002

Test Case Title: Verify that search results are relevant and accurately match the search criteria

Test Case Description: This test case verifies that search results are relevant and accurately match the search criteria entered by users.

Test Suite: Recipe Discovery

Test Priority: High

Preconditions:

- User is logged in or on the public search page
- Recipes with various keywords, ingredients, and cuisines have been added to the system

Test Data: Keywords, ingredients, and cuisine types

Test Steps:

1. Navigate to the search bar
2. Enter a combination of keywords, ingredients, and cuisine types in the search bar
3. Click the 'Search' button or press Enter

Postconditions:

- User is presented with a list of recipes related to the entered search criteria

Expected Result: Recipes related to the entered search criteria are displayed

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-RD-003

Test Case Title: Verify that users can filter search results by cuisine type, dietary preferences, and other criteria

Test Case Description: This test case verifies that users can filter search results by cuisine type, dietary preferences, and other criteria.

Test Suite: Recipe Discovery

Test Priority: Medium

Preconditions:

- User is logged in or on the public search page
- Recipes with various cuisine types, dietary preferences, and other criteria have been added to the system

Test Data: Cuisine type, dietary preference, and other criteria

Test Steps:

1. Navigate to the search page
2. Enter a keyword or ingredient in the search bar
3. Click the 'Search' button or press Enter
4. Apply filters using cuisine type, dietary preference, and other criteria

Postconditions:

- User is presented with a filtered list of recipes based on the chosen criteria

Expected Result: Recipes that meet the chosen filter criteria are displayed

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-RD-004

Test Case Title: Verify that recipes in the search results are accessible and detailed

Test Case Description: This test case verifies that recipes in the search results are accessible and contain detailed information.

Test Suite: Recipe Discovery

Test Priority: Medium

Preconditions:

- User is logged in or on the public search page
- Recipes with detailed information have been added to the system

Test Data: No test data needed

Test Steps:

1. Navigate to the search page
2. Enter a keyword or ingredient in the search bar
3. Click the 'Search' button or press Enter

4. Select a recipe from the search results

Postconditions:

- User is redirected to the selected recipe page
- Recipe page contains detailed information (ingredients, preparation steps, cook time, etc.)

Expected Result: Recipe pages are accessible and contain detailed information

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-RD-005

Test Case Title: Verify that users can save search results for future reference

Test Case Description: This test case verifies that users are able to save search results for future reference.

Test Suite: Recipe Discovery

Test Priority: Low

Preconditions:

- User is logged in
- Recipes are present in the search results

Test Data: Saved searches

Test Steps:

1. Navigate to the search page
2. Enter a keyword or ingredient in the search bar
3. Click the 'Search' button or press Enter
4. Save the search results using the 'Save Search' button
5. Access saved searches from the user profile

Postconditions:

- Search results can be accessed from the user's saved searches

Expected Result: Users can save and access search results for future reference

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive
