

# 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 ID: TC001

Test Case Title: Verify that users can add their name to their profile

Test Case Description: This test case verifies if a user is able to add their name while creating a new profile or editing an existing one.

Test Suite: User Profile Suite

Test Priority: High

Preconditions:

- User is logged into the social media platform
- User is on the profile editing page

Test Data: Any valid name (e.g., John Doe)

Test Steps:

1. Navigate to the profile editing page.
2. Enter a valid name in the 'Name' field.
3. Click the 'Save' button.

Postconditions:

- Name is displayed on the user's profile

Expected Result: User's name is correctly displayed on their profile.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that users can edit their bio

Test Case Description: This test case confirms if a user can create and modify their bio while creating a new profile or editing an existing one.

Test Suite: User Profile Suite

Test Priority: High

Preconditions:

- User is logged into the social media platform
- User is on the profile editing page

Test Data: Any valid bio text (e.g., "Software Engineer at XYZ")

Test Steps:

1. Navigate to the profile editing page.
2. Enter or modify the bio text in the 'Bio' field.
3. Click the 'Save' button.

Postconditions:

- Bio is displayed on the user's profile

Expected Result: User's bio is correctly displayed on their profile.

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 privacy settings for their profile picture

Test Case Description: This test case checks if a user can set privacy settings for their profile picture.

Test Suite: User Profile Suite

Test Priority: Medium

Preconditions:

- User is logged into the social media platform
- User has uploaded a profile picture

Test Data: N/A

Test Steps:

1. Navigate to the profile editing page.

2. Click on the profile picture to access the privacy settings.
3. Choose a privacy option for the profile picture (e.g., 'Public', 'Friends only', 'Private').
4. Click the 'Save' button.

Postconditions:

- Selected privacy setting is applied to the profile picture

Expected Result: User's chosen privacy setting is correctly applied to their profile picture.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that standard image formats are supported for the profile picture

Test Case Description: This test case verifies that standard image formats (JPG, JPEG, PNG) are supported for the profile picture.

Test Suite: User Profile Suite

Test Priority: Low

Preconditions:

- User is logged into the social media platform

Test Data: Profile picture in JPG, JPEG, and PNG format

Test Steps:

1. Navigate to the profile editing page.
2. Upload a profile picture in one of the supported formats.
3. Click the 'Save' button.

Postconditions:

- Profile picture is displayed on the user's profile

Expected Result: User's profile picture is displayed correctly, regardless of the supported format.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that a confirmation message is displayed upon saving the profile

Test Case Description: This test case ensures that a confirmation message is shown when a user successfully saves their profile.

Test Suite: User Profile Suite

Test Priority: Medium

Preconditions:

- User is logged into the social media platform
- User is on the profile editing page

Test Data: N/A

Test Steps:

1. Navigate to the profile editing page.
2. Enter or modify any profile information.
3. Click the 'Save' button.

Postconditions:

- Profile information is saved
- Confirmation message is displayed

Expected Result: Confirmation message is displayed upon successfully saving the user's profile.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

## **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 ID: TC-001

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

Test Case Description: This test case aims to ensure that the news feed correctly displays posts from accounts that the user has followed.

Test Suite: News Feed Module

Test Priority: High

Preconditions:

- User is logged in
- User has followed at least one account

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Navigate to the news feed page.
3. Verify that the user interface displays posts from the accounts the user has followed.

Postconditions:

- User remains logged in

Expected Result: News feed displays posts from accounts the user has followed.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC-002

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

Test Case Description: This test case aims to verify that the news feed displays content in the correct order, either chronologically or based on an algorithm.

Test Suite: News Feed Module

Test Priority: High

Preconditions:

- User is logged in
- User has followed at least two accounts that have posted content

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Navigate to the news feed page.
3. Verify that the content on the news feed is arranged in a chronological or algorithmic order.

Postconditions:

- User remains logged in

Expected Result: Content on the news feed is displayed in the correct order.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC-003

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

Test Case Description: This test case aims to validate that users can interact with posts on the news feed by liking, commenting, and sharing them.

Test Suite: News Feed Module

Test Priority: High

Preconditions:

- User is logged in



- User has followed at least one account that has posted content

Test Data: Prepare test data for interactions (e.g., liking, commenting, sharing)

Test Steps:

1. Log in as a registered user.
2. Navigate to the news feed page.
3. Select a post from the news feed.
4. Like the post.
5. Comment on the post.
6. Share the post.

Postconditions:

- User remains logged in
- The selected post has been liked, commented on, and shared

Expected Result: Users can interact with posts on the news feed.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC-004

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

Test Case Description: This test case aims to ensure that the news feed updates in real-time, reflecting new posts from followed accounts instantly.

Test Suite: News Feed Module

Test Priority: High

Preconditions:

- User is logged in
- User has followed at least two accounts that have posted content

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Navigate to the news feed page.
3. Monitor the news feed for a few minutes while the followed accounts post new content.
4. Verify that new posts from followed accounts appear on the news feed in real-time.

Postconditions:

- User remains logged in

Expected Result: News feed updates in real-time.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC-005

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

Test Case Description: This test case aims to validate that users can filter content on the news feed based on type or topic.

Test Suite: News Feed Module

Test Priority: High

Preconditions:

- User is logged in
- User has followed at least three accounts that have posted content of different types or topics

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Navigate to the news feed page.
3. Use available filters to narrow down the content based on type or topic.
4. Verify that the news feed only displays the filtered content.

Postconditions:

- User remains logged in

Expected Result: Users can filter content based on type or topic on the news feed.

Severity: Major

Priority: High

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 ID: TC-001

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

Test Case Description: This test case aims to validate the functionality of sending text messages between two users in a private conversation.

Test Suite: Messaging

Test Priority: High

Preconditions:

- User A and User B are registered on the social media platform
- User A and User B are friends

Test Data: No test data needed

Test Steps:

1. Log in as User A
2. Navigate to the messaging feature
3. Select User B from the friend list
4. Type a message in the text input field
5. Click on the 'Send' button

Postconditions:

- The message should appear in User B's inbox
- User B should receive a notification of the new message

Expected Result: The message is successfully delivered, and both users can view and respond to the message.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC-002

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

Test Case Description: This test case checks if users can send images as attachments in a private conversation.

Test Suite: Messaging

Test Priority: High

Preconditions:

- User A and User B are registered on the social media platform
- User A and User B are friends

Test Data: An image file for attachment

Test Steps:

1. Log in as User A
2. Navigate to the messaging feature
3. Select User B from the friend list
4. Click on the 'Attach' button to upload an image
5. Select the image file and click 'Open'
6. Click on the 'Send' button

Postconditions:

- The image appears in User B's inbox with a thumbnail preview
- User B should receive a notification of the new message

Expected Result: The image is successfully delivered, and both users can view and download the image.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC-003

Test Case Title: Verify that users can start group chats with multiple users

Test Case Description: This test case validates the ability for users to create and participate in group chats with more than two users.

Test Suite: Messaging

Test Priority: Medium

Preconditions:

- User A, User B, and User C are registered on the social media platform
- User A, User B, and User C are friends

Test Data: No test data needed

Test Steps:

1. Log in as User A
2. Navigate to the messaging feature
3. Click on the 'New Group Chat' button
4. Add User B and User C to the group chat
5. Type a message in the text input field

6. Click on the 'Send' button

Postconditions:

- The message is delivered to all group members
- Users B and C are displayed as group participants

Expected Result: Group chat is successfully created, and all group members can view, respond, and leave the group chat.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC-004

Test Case Title: Verify that messages show read receipts

Test Case Description: This test case checks if messages show read receipts, indicating when the recipient has viewed the message.

Test Suite: Messaging

Test Priority: Medium

Preconditions:

- User A and User B are registered on the social media platform
- User A and User B are friends

Test Data: No test data needed

Test Steps:

1. Log in as User A
2. Navigate to the messaging feature
3. Select User B from the friend list

4. Type a message in the text input field
5. Send the message and wait for User B to view it

Postconditions:

- The message status should update from 'Sent' to 'Read'

Expected Result: The message status changes from 'Sent' to 'Read' when the recipient views the message.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC-005

Test Case Title: Verify that users are notified of new messages

Test Case Description: This test case checks whether users receive notifications for new messages in private conversations or group chats.

Test Suite: Messaging

Test Priority: Medium

Preconditions:

- User A and User B are registered on the social media platform
- User A and User B are friends

Test Data: No test data needed

Test Steps:

1. Log in as User A
2. Navigate to the messaging feature
3. Select User B from the friend list or a group chat
4. Type a message in the text input field and send it



Postconditions:

- User B receives a notification for the new message

Expected Result: User B receives a notification for the new message, either in the app or via email/SMS.

Severity: Major

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 1:

Test Case ID: TC-CSS-001

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

Test Case Description: Validate the ability for users 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 post creation page

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

Test Steps:

1. Navigate to the post creation page.
2. Enter the sample text into the post input field.
3. Choose whether to share the post with followers or publicly.
4. Click the 'Share' button.

Postconditions:

- Post is visible on the user's profile
- Post is visible in the news feed of followers (if shared with followers)

Expected Result: Text post is successfully created and shared.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-CSS-002

Test Case Title: Verify that users can share photos

Test Case Description: Validate the ability for users to 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 post creation page

Test Data: Sample image file

Test Steps:

1. Navigate to the post creation page.
2. Click the 'Add Image' button.
3. Select the sample image file for upload.
4. Choose whether to share the post with followers or publicly.
5. Click the 'Share' button.

Postconditions:

- Uploaded photo is visible on the user's profile
- Uploaded photo is visible in the news feed of followers (if shared with followers)

Expected Result: Uploaded photo is successfully shared.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-CSS-003

Test Case Title: Verify that users can delete their posts

Test Case Description: Test the ability to delete a user's own text or image posts.

Test Suite: Content Sharing Suite

Test Priority: High

Preconditions:

- User is logged in
- User has at least one text or image post

Test Steps:

1. Navigate to the post on the user's profile.
2. Click the 'Delete' button.
3. Confirm deletion when prompted.

Postconditions:

- Deleted post is no longer visible on the user's profile
- Deleted post is no longer visible in the news feed of followers

Expected Result: Post is successfully deleted.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Negative

Test Case 4:

Test Case ID: TC-CSS-004

Test Case Title: Verify that users can use hashtags and tagging in posts

Test Case Description: Ensure that users can include hashtags and tag other users in posts, with proper functionality.

Test Suite: Content Sharing Suite

Test Priority: High

Preconditions:

- User is logged in
- User has access to a post creation page

Test Data: Text post with hashtags and tags

Test Steps:

1. Navigate to the post creation page.
2. Create a text post containing hashtags and tags (e.g., "#example @user").
3. Share the post (with followers or publicly).

Postconditions:

- Post is visible on the user's profile with hashtags and tags
- Clicking on a hashtag filters the news feed for posts containing the same hashtag
- Clicking on a tag notifies the tagged user

Expected Result: Hashtags and tags function as intended in the post.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC-CSS-005

Test Case Title: Verify that users receive feedback through post engagement

Test Case Description: Validate that users receive feedback on their posts through likes, comments, and shares.

Test Suite: Content Sharing Suite

Test Priority: High

Preconditions:

- User is logged in
- User has a post on their profile

Test Data: New like, comment, or share on user's post

Test Steps:

1. Log in as a different user.
2. Like, comment on, or share the original user's post.
3. Navigate to the original user's profile.

Postconditions:

- Post shows the updated like, comment, or share count (if applicable)
- Post displays the new like, comment, or share (if applicable)

Expected Result: Users receive feedback through likes, comments, and shares.

Severity: Major

Priority: High

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 ID: PRIV-001

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

Test Case Description: This test case aims to verify that users are able to choose and set their profile visibility to public, private, or restricted through the privacy settings feature.

Test Suite: Privacy Settings Suite

Test Priority: High

Preconditions:

- User is logged in to the social media platform.
- User has access to privacy settings.

Test Data: No test data needed

Test Steps:

1. Log in to the social media platform.
2. Navigate to the privacy settings page.
3. Locate the 'Profile Visibility' section.
4. Select 'Public', 'Private', or 'Restricted' as the desired profile visibility.
5. Save the changes made.

Postconditions:

- User profile visibility has been updated.
- The system displays a confirmation message indicating successful update.

Expected Result:

- User profile becomes publicly visible when set to 'Public'.
- User profile becomes invisible to non-friends when set to 'Private'.
- User profile becomes only visible to approved friends when set to 'Restricted'.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: PRIV-002

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

Test Case Description: This test case aims to verify that users can modify their privacy settings at any given point in time.

Test Suite: Privacy Settings Suite

Test Priority: Medium

Preconditions:

- User is logged in to the social media platform.
- User has access to privacy settings.

Test Data: No test data needed



#### Test Steps:

1. Log in to the social media platform.
2. Navigate to the privacy settings page.
3. Modify the current profile visibility setting.
4. Save the changes made.
5. Repeat steps 2-4 multiple times with different privacy settings.

#### Postconditions:

- User profile visibility has been updated to the new chosen setting.
- The system displays a confirmation message indicating successful update.

#### Expected Result:

- The system allows users to modify privacy settings repeatedly.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: PRIV-003

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

Test Case Description: This test case aims to verify that users can change the visibility of individual posts to public, private, or friends only through the post settings.

Test Suite: Privacy Settings Suite

Test Priority: High

#### Preconditions:

- User is logged in to the social media platform.
- User has access to privacy settings and post creation feature.

Test Data: Multiple test posts with different content, created for testing purposes.

#### Test Steps:

1. Log in to the social media platform.
2. Create a new post with some content.
3. Before publishing, locate and click on the post's visibility settings.
4. Select 'Public', 'Private', or 'Friends Only'.
5. Publish the post.
6. Repeat steps 2-5 for multiple test posts with different visibility settings.

Postconditions:

- Test posts have been published with the chosen visibility settings.

Expected Result:

- Individual posts become visible to the public, only to the user, or to friends based on the chosen setting.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: PRIV-004

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

Test Case Description: This test case aims to verify that the system immediately adheres to the user's chosen privacy settings without any delays or errors.

Test Suite: Privacy Settings Suite

Test Priority: High

Preconditions:

- User is logged in to the social media platform.
- User has access to privacy settings and post creation feature.
- User has set a specific visibility setting.

Test Data: Test posts with different content, created for testing purposes.

Test Steps:

1. Log in to the social media platform.
2. Create a new test post.
3. Publish the test post.
4. Verify that the test post is visible according to the chosen visibility setting.
5. Repeat steps 2-4 for multiple test posts with different visibility settings.

Postconditions:

- Test posts have been published and are immediately visible according to user's chosen settings.

Expected Result:

- The system respects the user's privacy settings and displays or hides posts based on the chosen visibility.

Severity: Blocker

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: PRIV-005

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

Test Case Description: This test case aims to verify that users receive a notification when their privacy settings have been changed by them or by someone else.

Test Suite: Privacy Settings Suite

Test Priority: Medium

Preconditions:

- User A is logged in to the social media platform.
- User A has access to privacy settings.
- User B is a friend of User A.

Test Data: No test data needed

Test Steps:

1. Log in to the social media platform as User A.

2. Change User A's privacy settings.
3. Log in as User B and navigate to User A's profile.
4. Verify if User B can or cannot view User A's profile or posts based on the updated privacy settings.
5. Observe if User A receives a notification regarding the privacy settings change.

Postconditions:

- User B can or cannot view User A's profile or posts based on the updated privacy settings.
- User A receives a notification regarding the privacy settings change.

Expected Result:

- Users receive a notification when privacy settings are changed.

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 Suite: Notifications

Test Priority: High

Preconditions:

- \* User has an active account
- \* User is logged in
- \* Another user has followed the first user and liked, commented, or shared their post

Test Data: Post with content, user data for following, liking, commenting, and sharing

Test Steps:

1. Log in as the primary user
2. Verify that the post is visible in the user's feed
3. Log in as the secondary user
4. Follow the primary user
5. Like, comment, and share the primary user's post
6. Return to the primary user's account and refresh the feed

Postconditions:

- \* Notification for likes, comments, and shares is visible on the primary user's account

Expected Result:

- \* The primary user receives notifications for the actions performed by the secondary user

Severity: Major

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 Suite: Notifications

Test Priority: High

Preconditions:

- \* User has an active account
- \* User is logged in
- \* Another user exists who has not yet connected with the primary user

Test Data: User data for the primary and secondary users

Test Steps:

1. Log in as the primary user
2. Log in as the secondary user
3. Navigate to the primary user's profile
4. Send a friend request or mention the primary user in a post
5. Return to the primary user's account and refresh the page

Postconditions:

- \* Friend request or mention notification is visible on the primary user's account

Expected Result:

\* The primary user receives notifications for new friend requests and mentions

Severity: Major

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 Suite: Notifications

Test Priority: High

Preconditions:

\* User has an active account

\* User is logged in

\* Another user has followed, liked, commented, or shared the primary user's post

Test Data: User data and post content

Test Steps:

1. Log in as the primary user

2. Log in as the secondary user

3. Follow, like, comment, or share a post from the primary user

4. Observe the notifications list on the primary user's account

Postconditions:

\* Notifications for the actions performed by the secondary user are immediately visible

Expected Result:

\* Notifications are displayed in real-time on the primary user's account

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 Suite: Notifications

Test Priority: Medium

Preconditions:

- \* User has an active account

- \* User is logged in

Test Data: Various notification options for customization

Test Steps:

1. Log in as the primary user
2. Navigate to the user settings
3. Select the notifications tab
4. Update the notification preferences
5. Perform various actions that would typically generate a notification

Postconditions:

- \* Notifications are displayed based on the updated preferences

Expected Result:

- \* The primary user can customize notification preferences, tailoring the notifications they receive

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 Suite: Notifications

Test Priority: Medium

Preconditions:

- \* User has an active account
- \* User is logged in
- \* Another user has liked, commented, or shared the primary user's post

Test Data: User data and post content

Test Steps:

1. Log in as the primary user
2. Log in as the secondary user
3. Follow, like, comment, or share a post from the primary user
4. View the notification and click the link

Postconditions:

- \* The link directs the primary user to the post or the user's profile

Expected Result:

- \* Notifications include links to the relevant activity, allowing users to quickly access the posts or profiles associated with the notifications

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

-----

**Data - 7**

**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 aims to validate the basic functionality of the search feature by verifying if users can search for content using keywords and hashtags.

Test Suite: Search Functionality

Test Priority: High

Preconditions:

\* User is logged in

\* Search bar is visible and accessible

Test Data: Keyword or hashtag related to existing content

Test Steps:

1. Navigate to the search bar
2. Input a keyword or hashtag in the search bar
3. Press enter or click the search button

Postconditions:

\* Search results are displayed

Expected Result: Relevant content containing the keyword or hashtag is displayed in the search results

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 checks if the search results contain various content types, including profiles, posts, and topics.

Test Suite: Search Functionality

Test Priority: High

Preconditions:

\* User is logged in

\* Search bar is visible and accessible

Test Data: Keyword or hashtag related to existing content (profiles, posts, and topics)

Test Steps:

1. Navigate to the search bar
2. Input a keyword or hashtag in the search bar related to profiles, posts, and topics

3. Press enter or click the search button

Postconditions:

- \* Search results are displayed

Expected Result: Profiles, posts, and topics containing the keyword or hashtag are displayed in the search results

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 search results are sorted by relevance, showing the most relevant content first.

Test Suite: Search Functionality

Test Priority: Medium

Preconditions:

- \* User is logged in

- \* Search bar is visible and accessible

Test Data: Keyword or hashtag with multiple relevant content pieces

Test Steps:

1. Navigate to the search bar
2. Input a keyword or hashtag in the search bar
3. Press enter or click the search button

Postconditions:

- \* Search results are displayed

Expected Result: The most relevant content containing the keyword or hashtag is displayed at the top of the

search results

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 checks if users can filter search results by content type, such as posts, profiles, or topics.

Test Suite: Search Functionality

Test Priority: Medium

Preconditions:

- \* User is logged in
- \* Search bar is visible and accessible

Test Data: Keyword or hashtag with multiple relevant content pieces across different types

Test Steps:

1. Navigate to the search bar
2. Input a keyword or hashtag in the search bar
3. Press enter or click the search button
4. Click or select the filter button
5. Choose a content type

Postconditions:

- \* Filter is applied

Expected Result: Search results are filtered to display only 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 checks if the search feature offers suggestions as the user inputs keywords or hashtags.

Test Suite: Search Functionality

Test Priority: Low

Preconditions:

- \* User is logged in
- \* Search bar is visible and accessible

Test Data: Keyword or hashtag input sequence

Test Steps:

1. Navigate to the search bar
2. Input a keyword or hashtag
3. Observe the suggestions provided

Postconditions:

- \* Suggestions are presented

Expected Result: Relevant suggestions based on the input keyword or hashtag are displayed in the search bar

Severity: Major

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-HAS-001

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

Test Case Description: This test case validates the ability to add hashtags to a user's post.

Test Suite: Hashtags and Tagging

Test Priority: High

Preconditions:

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

Test Data: "This is a post with #hashtag and #testtag"

Test Steps:

1. Log in as a user
2. Navigate to the post creation screen
3. Enter test data as the post content
4. Include hashtags in the post
5. Submit the post

Postconditions:

- Post is published with hashtags

Expected Result:

- The post is published with hashtags "#hashtag" and "#testtag" that are clickable and searchable.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-HAS-002

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

Test Case Description: This test case evaluates the user's ability to tag other profiles, brands, or locations in their posts.

Test Suite: Hashtags and Tagging

Test Priority: High

Preconditions:



- User is logged in
- User has the ability to create a new post
- At least one other user or brand exists on the platform

Test Data: "This is a post tagging @user, @brand, and location: #city"

Test Steps:

1. Log in as a user
2. Navigate to the post creation screen
3. Enter test data as the post content
4. Include tags for other users or brands and a location
5. Submit the post

Postconditions:

- Post is published with tags

Expected Result:

- The post is published with tags for "@user", "@brand", and "location: #city" that trigger notifications for the tagged users.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-HAS-003

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

Test Case Description: This test case certifies that clickable hashtags and tags direct users to related content.

Test Suite: Hashtags and Tagging

Test Priority: High

Preconditions:

- User is logged in
- Posts with hashtags and tags have been published

Test Data: Hashtags and tags present in published posts

Test Steps:

1. Log in as a user
2. Navigate to the feed or search for a hashtag or tag
3. Click on a hashtag or tag
4. View the content related to the hashtag or tag

Postconditions:

- Displayed content is relevant to the hashtag or tag

Expected Result:

- Users are taken to a page with related content when clicking on hashtags or tags.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-HAS-004

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

Test Case Description: This test case confirms the ability to search for and find posts using hashtags.

Test Suite: Hashtags and Tagging

Test Priority: Medium

Preconditions:

- User is logged in
- Posts with hashtags have been published

Test Data: Existing hashtags

#### Test Steps:

1. Log in as a user
2. Navigate to the search bar
3. Search using a hashtag
4. View the search results

#### Postconditions:

- Search results display posts with the entered hashtag

#### Expected Result:

- Search results provide related posts for the entered hashtag.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

#### Test Case 5:

Test Case ID: TC-HAS-005

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

Test Case Description: This test case checks the system's ability to suggest popular hashtags based on user input.

Test Suite: Hashtags and Tagging

Test Priority: Low

#### Preconditions:

- User is logged in
- Post creation screen is accessible

Test Data: User-entered text in the post input field

#### Test Steps:

1. Log in as a user

2. Navigate to the post creation screen

3. Type text in the input field

4. Check for hashtag suggestions

Postconditions:

- System provides relevant hashtag suggestions

Expected Result:

- System generates a list of popular hashtags based on the user's input that can be added to the post.

Severity: Minor

Priority: Low

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 1:

Test Case ID: TC-001\_EMJ\_RACT\_PT-1

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

Test Case Description: As a user, I want to be able to react to a post using various emojis provided by the platform to express my feelings towards the content.

Test Suite: Emojis and Reactions

Test Priority: High

Preconditions:

1. User is logged in.
2. A post exists on the user's news feed.

Test Data: A test post and a variety of emojis available on the platform.

Test Steps:

1. Navigate to the user's news feed.
2. Identify a test post.
3. Select the emoji reaction icon.
4. Choose a desired emoji from the list.
5. Confirm the selected emoji has been applied to the post.

Postconditions:

1. The chosen emoji reaction is visible under the post.
2. The reaction count is updated accordingly.

Expected Result: Users can react to posts with a variety of emojis and the platform supports multiple emoji reactions for each post.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-002\_EMJ\_CMNT\_PT-2

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

Test Case Description: As a user, I want to be able to incorporate emojis within my comments and messages to make the communication more engaging and expressive.

Test Suite: Emojis and Reactions

Test Priority: Medium

Preconditions:

1. User is logged in.
2. A post or conversation exists.

Test Data: Test posts, comments, or messages and a variety of emojis available on the platform.

Test Steps:

1. Navigate to a test post.
2. Enter a comment and include an emoji.
3. Submit the comment.
4. Verify if the emoji appears within the comment.
5. Repeat the above steps for a conversation or direct message.

Postconditions:

1. Emoji is displayed within the comment.
2. Emoji is displayed within the message.

Expected Result: Emojis can be used in comments and messages, enhancing the user's ability to express themselves.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-003\_EMJ\_RCNT\_PT-3

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

Test Case Description: As a user, I want to be able to quickly see the number of reactions a post has received, providing an overview of other users' feedback.

Test Suite: Emojis and Reactions

Test Priority: High

Preconditions:

1. User is logged in.
2. A post with applied reactions exists on the user's news feed.

Test Data: Test post with multiple user reactions.

Test Steps:

1. Navigate to the user's news feed.
2. Identify a test post with reactions.
3. Check if the reaction count is displayed under the post.
4. Verify if the reaction count is consistent with the number of applied reactions.

Postconditions: Reaction count is visible and accurate.

Expected Result: Users can see a count of reactions on posts, reflecting the number of users that have reacted

to the content.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Visual Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-004\_WIDE\_EMJ\_PT-4

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

Test Case Description: As a user, I want the platform to support various emoji standards, ensuring compatibility with different devices and applications.

Test Suite: Emojis and Reactions

Test Priority: Medium

Preconditions:

1. User is logged in.
2. Access to a wide range of emojis from different platforms and devices.

Test Data: Emojis from diverse platforms and sources.

Test Steps:

1. Copy emojis from various sources.
2. Paste them into a comment, message, or reaction field.
3. Verify if the platform correctly displays and processes the copied emojis.

Postconditions:

1. Platform supports and displays emojis from various sources.

Expected Result: The platform supports a wide range of emojis, ensuring compatibility with different devices and operating systems.

Severity: Major

Priority: Medium



Type of Testing: Functional Testing, Compatibility Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC-005\_REM\_CHG\_RC\_PT-5

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

Test Case Description: As a user, I want to be able to remove or modify my reactions on posts if I have made a mistake or change my mind about expressing my feelings.

Test Suite: Emojis and Reactions

Test Priority: High

Preconditions:

1. User is logged in.
2. A post with the user's reaction exists on the user's news feed.

Test Data: Test post with the user's reaction.

Test Steps:

1. Navigate to the user's news feed.
2. Identify a test post with the user's reaction.
3. Select the user's reaction.
4. Modify or remove the user's reaction.
5. Confirm the change or removal was successful.

Postconditions:

1. User reaction has been removed or changed.

Expected Result: Users can remove or change their reactions on posts, allowing them to correct mistakes or adjust their expressed feelings.

Severity: Major

Priority: High

Type of Testing: Functional Testing

## **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: TC-001

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

Test Case Description: As a user, I want to successfully start and stop a live stream to ensure the basic functionality of the live streaming feature.

Test Suite: Live Streaming Module

Test Priority: High

Preconditions:

- User is logged in.
- User has a stable internet connection.
- User has permissions to live stream.

Test Data: No test data needed

Test Steps:

1. Log in to the social media platform.
2. Navigate to the live streaming section.
3. Click on the "Start Live Stream" button.
4. Wait for 1-2 minutes while streaming.
5. Click on the "Stop Live Stream" button.

Postconditions:

- The live stream is no longer visible.
- The recorded live stream is saved for later viewing.

Expected Result: User can start and stop live streams without any issues.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC-002

Test Case Title: Verify that live streams can be public or limited to followers.

Test Case Description: As a user, I want to be able to choose between making my live stream public or available only to my followers to ensure the privacy settings work as intended.

Test Suite: Live Streaming Module

Test Priority: High

Preconditions:

- User is logged in.
- User has a stable internet connection.
- User has permissions to live stream.

Test Data: No test data needed

Test Steps:

1. Log in to the social media platform.
2. Navigate to the live streaming section.
3. Set the privacy settings for the live stream to "Public".
4. Start the live stream.
5. Repeat steps 2-4, this time setting the privacy settings to "Followers only".

Postconditions:

- The public live stream is visible to all users.
- The followers-only live stream is only visible to the user's followers.

Expected Result: Privacy settings for live streams function as expected.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC-003

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

Test Case Description: As a user watching a live stream, I want to be able to leave comments and use reactions during the live stream to engage with the content and the streamer.

Test Suite: Viewer Interaction Module

Test Priority: Medium

Preconditions:

- User is logged in.
- User has a stable internet connection.
- User is watching an active live stream.

Test Data: No test data needed

Test Steps:

1. Log in to the social media platform.
2. Navigate to a live stream.
3. Enter a comment in the comment box.
4. Click on a reaction icon.

Postconditions:

- The comment is displayed under the live stream.
- The reaction count increases, and the reaction is shown beside the user's name in the comments.

Expected Result: User can interact with live stream through comments and reactions.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC-004

Test Case Title: Verify that the stream displays the viewer count and live comments.

Test Case Description: As a user watching a live stream, I want to see the viewer count and live comments for an engaging and interactive experience.

Test Suite: Viewer Interaction Module

Test Priority: Medium

Preconditions:

- User is logged in.
- User has a stable internet connection.
- User is watching an active live stream.

Test Data: No test data needed

Test Steps:

1. Log in to the social media platform.
2. Navigate to a live stream.
3. Observe the viewer count and live comments.

Postconditions:

- The viewer count is accurately displayed and updates in real-time.
- Live comments are displayed under the live stream in real-time.

Expected Result: The stream displays the viewer count and live comments.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC-005

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

Test Case Description: As a user, I want to save a live stream for later viewing to ensure that I can revisit and enjoy the content at my convenience.

Test Suite: Live Streaming Module

Test Priority: High

Preconditions:

- User is logged in.
- User has a stable internet connection.
- User is watching a live stream.

Test Data: No test data needed

Test Steps:

1. Log in to the social media platform.
2. Navigate to a live stream.
3. Click on the "Save for Later" button.

Postconditions:

- The live stream is saved in the user's "Saved" section.

Expected Result: Users can save live streams for later viewing.

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 ID: TC-001

Test Case Title: Verify that users can create stories with photos, videos, and text overlays

Test Case Description: As a user, I want to create a story by uploading a photo, recording a video, or adding text overlays to share temporary moments with my followers.

Test Suite: Story Creation

Test Priority: High

Preconditions:



1. User is logged in
2. User navigates to the story creation screen

Test Data: A photo, a video, and text for the overlay

Test Steps:

1. Tap on the "Create Story" button
2. Select "Add Photo" and choose an image from the gallery
3. Add text overlay by tapping "Add Text"
4. Save the story
5. Repeat steps 1-4, but this time select "Add Video" and record a video

Expected Result: The story is successfully created with the selected photo, video, or text overlay, and the user is directed to the story viewing screen.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-002

Test Case Title: Verify that stories disappear after 24 hours

Test Case Description: As a user, I want my stories to automatically disappear after 24 hours so that they do not clutter my followers' feeds.

Test Suite: Story Expiry

Test Priority: High

Preconditions:

1. User is logged in
2. User has an active story

Test Data: No test data needed

Test Steps:

1. Create a story
2. Wait for 24 hours
3. Check if the story is still visible in the user's or any follower's story list

Expected Result: The story is no longer visible in the user's or any follower's story list after 24 hours.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Time-based

Test Case ID: TC-003

Test Case Title: Verify that users can see who viewed their stories

Test Case Description: As a user, I want to see who has viewed my stories to understand the engagement and reach of my shared moments.

Test Suite: Story Viewing Tracking

Test Priority: Medium

Preconditions:

1. User is logged in
2. User has an active story

Test Data: No test data needed

Test Steps:

1. Create a story
2. Wait for some followers to view the story
3. Tap on the story to view it
4. Check the list of viewers

Expected Result: The list of viewers includes the followers who have viewed the story.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-004

Test Case Title: Verify that stories can be shared publicly or with selected followers

Test Case Description: As a user, I want to control the visibility of my stories by sharing them publicly or with a specific group of followers.

Test Suite: Story Privacy Settings

Test Priority: Medium

Preconditions:

1. User is logged in
2. User has at least two groups of followers (e.g., friends and family)

Test Data: No test data needed

Test Steps:

1. Create a story
2. Select the privacy settings for the story (Public or Specific Groups)
3. Save the story
4. Verify that the story is visible to the selected audience (Public or Specific Groups)

Expected Result: The story is only visible to the chosen audience (Public or Specific Groups).

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-005

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

Test Case Description: As a user, I want to customize my stories with stickers and filters to make them more

engaging and visually appealing.

Test Suite: Story Customization

Test Priority: Low

Preconditions:

1. User is logged in
2. User is on the story creation screen

Test Data: No test data needed

Test Steps:

1. Create a story
2. Tap on the stickers or filters icon
3. Select a sticker or filter and apply it to the story
4. Save the story

Expected Result: The story now includes the selected sticker or filter, and the user is directed to the story viewing screen.

Severity: Minor

Priority: Low

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: TC-001

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 followers, such as their location, age range, and gender.

Test Suite: Analytics and Insights

Test Priority: High

Preconditions:

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

Test Data: N/A

Test Steps:

1. Log in to the social media platform as a user.
2. Navigate to the Analytics and Insights section.
3. Select the Follower Demographics tab.
4. Observe the displayed data, which should include location, age range, and gender of followers.

Postconditions: N/A

Expected Result: The system displays accurate follower demographics data in a user-friendly format.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-002

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

Test Case Description: As a user, I want to view the reach and engagement metrics of my posts, such as likes, comments, shares, and impressions.

Test Suite: Analytics and Insights

Test Priority: High

Preconditions:

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

Test Data: N/A

Test Steps:

1. Log in to the social media platform as a user.
2. Navigate to the Analytics and Insights section.
3. Select the Post Reach and Engagement tab.

4. Observe the displayed data, which should include likes, comments, shares, and impressions for each post.

Postconditions: N/A

Expected Result: The system displays accurate post reach and engagement metrics in a user-friendly format.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-003

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 data to be presented in a clear, easy-to-understand format.

Test Suite: Analytics and Insights

Test Priority: Medium

Preconditions:

- User is logged in

Test Data: N/A

Test Steps:

1. Log in to the social media platform as a user.
2. Navigate to the Analytics and Insights section.
3. Observe the layout, design, and overall readability of the displayed data.

Postconditions: N/A

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

Severity: Major

Priority: Medium

Type of Testing: Usability Testing

Test Case Approach: Positive

Test Case ID: TC-004

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

Test Case Description: As a user, I want to filter the analytics and insights data by specific time ranges and content types.

Test Suite: Analytics and Insights

Test Priority: Medium

Preconditions:

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

Test Data: N/A

Test Steps:

1. Log in to the social media platform as a user.
2. Navigate to the Analytics and Insights section.
3. Apply filters for time range (e.g., last 7 days) and content type (e.g., photos).
4. Observe the updated analytics and insights data.

Postconditions: N/A

Expected Result: The system updates the analytics and insights data based on the applied filters for time range and content type.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-005

Test Case Title: Verify that invalid filters do not break functionality



Test Case Description: As a user, I want the system to handle cases where invalid filters are applied, ensuring that the analytics and insights page remains functional.

Test Suite: Analytics and Insights

Test Priority: Low

Preconditions:

- User is logged in

Test Data: N/A

Test Steps:

1. Log in to the social media platform as a user.
2. Navigate to the Analytics and Insights section.
3. Apply invalid filters (e.g., non-existent time ranges, content types not available to the user).
4. Observe the displayed analytics and insights data.

Postconditions: N/A

Expected Result: The system handles invalid filters gracefully, displaying appropriate error messages, if needed, but keeps the analytics and insights page functional and usable.

Severity: Minor

Priority: Low

Type of Testing: Negative

Test Case Approach: Negative

**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: TC001

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

Test Case Description: Validate the ability for a user to search for and successfully join an interest-based group on the social media platform.

Test Suite: Group Management

Test Priority: High

Preconditions:

- User is logged in
- User is on the groups page or group directory

Test Data: A pre-existing interest-based group that allows new members

#### Test Steps:

1. Navigate to the groups page
2. Use the search bar to find a specific interest-based group
3. Click on the group's name to view its details
4. Click the "Join Group" button
5. Confirm the action in the subsequent pop-up

#### Postconditions:

- User is a member of the group
- User is redirected to the group's page

Expected Result: Group join request is successful, and the user becomes a member of the group.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that event details include date, time, location, and description

Test Case Description: Validate the accurate display of event information when creating or viewing events.

Test Suite: Events Management

Test Priority: High

#### Preconditions:

- User is logged in
- User has created an event or there is a preexisting event

Test Data: A created event or a preexisting event

#### Test Steps:

1. Navigate to the events page
2. Click on the desired event

3. Check that the event displays the date, time, location, and description

Postconditions:

- Event information is visible and accurate

Expected Result: All required event details are displayed correctly, allowing users to obtain necessary information about the event.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users can RSVP to events and see the list of attendees

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

Test Suite: Events Management

Test Priority: High

Preconditions:

- User is logged in
- User is on the event page
- Event allows RSVPs

Test Data: An event that allows RSVPs

Test Steps:

1. Navigate to the events page
2. Click on the desired event that allows RSVPs
3. Click the "RSVP" button
4. Confirm the action in the subsequent pop-up
5. Verify that the user's RSVP status is displayed (e.g., attending, interested)
6. Refresh the page and verify that the user's RSVP status appears in the list of attendees

Postconditions:

- User's RSVP status is recorded and displayed
- User appears in the list of attendees

Expected Result: User successfully RSVPs to an event, and the RSVP status is displayed correctly for the user and within the list of attendees.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that group discussions can be organized by topic

Test Case Description: Validate the ability for group admins or members to create topics for discussions within the group.

Test Suite: Group Discussions

Test Priority: Medium

Preconditions:

- User is logged in
- User is a member of a group

Test Data: A group with discussion permissions enabled

Test Steps:

1. Navigate to the desired group
2. Click the "Discussions" tab
3. Click the "New Topic" button
4. Enter a topic title and description
5. Post the topic
6. Verify that the new topic appears in the group's discussions

Postconditions:

- New discussion topic is visible within the group

Expected Result: Group members can create, view, and engage with discussion topics related to the group's interests or purpose.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify users can create and join events

Test Case Description: Validate the ability for a user to create an event and join an event created by another user.

Test Suite: Events Management

Test Priority: High

Preconditions:

- User is logged in

Test Data: An event created by the user and an event created by another user

Test Steps:

1. Navigate to the events page
2. Click the "Create Event" button
3. Enter event title, date, time, location, and description
4. Save the event
5. Verify that the event appears on the events page
6. Click on the event created by another user
7. Click the "Join Event" button
8. Confirm the action in the subsequent pop-up

Postconditions:

- User-created event is visible on the events page
- User has joined an event created by another user

Expected Result: Users can successfully create and join events, allowing them to connect with others and engage in activities related to shared interests.

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 Options

Test Priority: High

Preconditions:

- User is logged into the social media platform
- User has a complete profile with verified email and phone number

Test Data: No test data needed

Test Steps:

1. Navigate to the account settings page
2. Click on the "Monetization" tab
3. Select the desired monetization option (e.g., advertising revenue)
4. Complete the application form with accurate information
5. Submit the application

Postconditions:

- User's monetization application status is displayed as "Pending Review"

Expected Result: The test passes if the user can access the monetization application form, complete it, and



submit it, resulting in a "Pending Review" status.

Severity: Major

Priority: High

Type of Testing: Functional Testing, End-to-End Testing, Usability 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 confirms that the monetization options include advertising revenue and sponsored content partnerships.

Test Suite: Monetization Options

Test Priority: Medium

Preconditions:

- User has access to monetization options and their application is approved

Test Data: N/A

Test Steps:

1. Log in as a content creator with approved monetization application
2. Navigate to the monetization settings page
3. Verify that "Advertising Revenue" and "Sponsored Content" are available as monetization options

Postconditions: N/A

Expected Result: The test passes if both "Advertising Revenue" and "Sponsored Content" are available as monetization options for the user.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, System 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 ensures that users can receive payments from their monetization activities through the social media platform.

Test Suite: Monetization Options

Test Priority: High

Preconditions:

- User has a complete profile with verified email, phone number, and payment information
- User has active monetization options with approved applications

Test Data: N/A

Test Steps:

1. Log in as a content creator with approved monetization application
2. Generate earnings by creating content and engaging users
3. Request a payout through the platform's payout system
4. Monitor the payout status until it's processed and completed

Postconditions: N/A

Expected Result: The test passes if the user can receive payments from their monetization activities through the platform.

Severity: Critical

Priority: High

Type of Testing: Functional Testing, System Testing, Regression 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 confirms that users have access to detailed earnings reports to track their monetization performance.

Test Suite: Monetization Options

Test Priority: Medium

Preconditions:

- User has active monetization options with earnings generated

Test Data: N/A

Test Steps:

1. Log in as a content creator with active monetization options and earnings
2. Navigate to the earnings report section
3. Verify that earnings by source, time period, and user engagement are available
4. Filter, sort, and analyze the earnings data

Postconditions: N/A

Expected Result: The test passes if detailed earnings reports are available for users to analyze their monetization performance.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, System 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 confirms that monetization eligibility requirements are clearly communicated and accessible to users.

Test Suite: Monetization Options

Test Priority: Low

Preconditions: N/A

Test Data: N/A

Test Steps:

1. Visit the social media platform's main page as an anonymous user
2. Navigate to the monetization options page or related content
3. Verify that the monetization eligibility requirements are listed and accessible

Postconditions: N/A

Expected Result: The test passes if the monetization eligibility requirements are easily accessible, clear, and detailed for users.

Severity: Minor

Priority

-----

**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: TC-EXP-001

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

Test Case Description: As a user, validate that the explore tab displays a collection of popular and trending content from across the platform.

Test Suite: Explore Tab Functionality

Test Priority: High

Preconditions:

- User is logged in.
- User is on the home page or news feed.

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Navigate to the explore or discover tab.
3. Observe the content displayed on the page.

Postconditions:

- User remains logged in.

Expected Result:

- The explore tab displays a collection of popular and trending content from across the platform.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-EXP-002

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

Test Case Description: As a user, confirm that content displayed on the explore tab is tailored to the user's interests and engagement history.

Test Suite: Explore Tab Functionality

Test Priority: High

Preconditions:

- User is logged in.
- User has a history of interactions (e.g., likes, comments, shares) on the platform.

Test Data: User interaction history, interests, and social connections

Test Steps:

1. Log in as a registered user.
2. Interact with various posts, photos, and videos on the platform.
3. Navigate to the explore or discover tab.
4. Observe the content displayed on the page.

Postconditions:

- User remains logged in.

Expected Result:

- Content displayed on the explore tab is aligned with the user's interests, engagement history, and social connections.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-EXP-003

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

Test Case Description: As a user, ensure that the explore tab provides the ability to filter content based on categories or topics.

Test Suite: Explore Tab Functionality

Test Priority: Medium

Preconditions:

- User is logged in.

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Navigate to the explore or discover tab.
3. Locate and engage with the filtering options (e.g., categories, topics).
4. Apply a filter to the content displayed on the page.

Postconditions:

- User remains logged in.

Expected Result:

- The explore tab allows users to filter content based on categories or topics, displaying relevant content.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-EXP-004

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

Test Case Description: As a user, validate that the explore tab is updated consistently with new and trending

content.

Test Suite: Explore Tab Functionality

Test Priority: Medium

Preconditions:

- User is logged in.
- User has previously viewed the explore tab.

Test Data: No test data needed

Test Steps:

1. Log in as a registered user.
2. Navigate to the explore or discover tab.
3. Observe the content displayed on the page.
4. Wait for a predefined period (e.g., 1 hour, 1 day).
5. Navigate back to the explore tab and observe any changes in the content displayed.

Postconditions:

- User remains logged in.

Expected Result:

- The explore tab is consistently updated with new and trending content.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-EXP-005

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

Test Case Description: As a user, ensure that the explore tab allows users to save or follow content for future reference.

Test Suite: Explore Tab Functionality



Test Priority: Medium

Preconditions:

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

Test Data: A post, photo, or video from the explore tab

Test Steps:

1. Log in as a registered user.
2. Navigate to the explore or discover tab.
3. Select a post, photo, or video to save or follow.
4. Engage with the save or follow option.
5. Verify that the selected content is saved or followed.

Postconditions:

- User remains logged in.

Expected Result:

- The explore tab allows users to save or follow content for future reference.

Severity: Major

Priority: Medium

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 ID: TC001

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

Test Case Description: This test case aims to ensure that the social media platform includes descriptive alternative text for all images, allowing users relying on screen readers to understand the content of the images.

Test Suite: Accessibility Features

Test Priority: High

Preconditions:

- User is logged in
- User navigates to a post containing an image

Test Data: No test data needed

Test Steps:

1. Log in as a user.
2. Navigate to a post containing an image.
3. Activate the screen reader tool.
4. Read the post with the screen reader.

Postconditions:

- Screen reader accurately reads out the alternate text

Expected Result:

- Screen reader reads out the descriptive alternate text accompanying the image

Severity: Major

Priority: High

Type of Testing: Functional Testing, Usability Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify screen reader compatibility with the platform

Test Case Description: This test case aims to ensure that the social media platform is fully compatible with screen readers, allowing users who rely on such tools to navigate and interact with the platform effectively.

Test Suite: Accessibility Features

Test Priority: High

Preconditions:

- User is logged in
- User has a screen reader tool installed and enabled

Test Data: No test data needed

Test Steps:

1. Log in as a user.

2. Activate the screen reader tool.

3. Navigate through the various sections of the platform, such as news feed, messaging, user profiles, etc.

4. Interact with different elements (posts, photos, buttons, etc.) on the platform

Postconditions:

- Screen reader accurately reads out all elements and interactions

Expected Result:

- User can navigate and interact with the platform using a screen reader without any barriers or limitations

Severity: Critical

Priority: High

Type of Testing: Functional Testing, Usability Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that keyboard shortcuts can be used for navigation

Test Case Description: This test case aims to ensure that users can navigate and interact with the platform using only the keyboard, allowing users with motor impairments to use the platform effectively.

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. Close the mouse and rely on the keyboard to navigate through the various sections of the platform, such as news feed, messaging, user profiles, etc.

3. Interact with different elements (posts, photos, buttons, etc.) on the platform without using the mouse

Postconditions:

- User can complete all tasks using only the keyboard

Expected Result:

- User successfully navigates and interacts with the platform using only keyboard shortcuts

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Usability Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that color contrast and font size options are provided

Test Case Description: This test case aims to ensure that the social media platform provides options for users to adjust color contrast and font size to meet their visual requirements and preferences, allowing users with visual impairments to use the platform effectively.

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 settings page.
3. Locate and adjust the color contrast and font size settings.
4. Navigate through the various sections of the platform and observe the changes in contrast and size.

Postconditions:

- The platform displays the adjusted contrast and size settings

Expected Result:

- User can adjust color contrast and font size according to their needs, and platform displays the changes

accordingly

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Usability Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that accessibility features are clearly documented

Test Case Description: This test case aims to ensure that the social media platform provides a clear and comprehensive document explaining all the accessibility features available, allowing users to understand and utilize these features effectively.

Test Suite: Accessibility

Test Priority: Low

Preconditions:

- User has access to platform documentation

Test Data: No test data needed

Test Steps:

1. Locate the platform documentation for accessibility features.
2. Read through the entire document.
3. Verify that all the accessibility features (alt text, screen reader compatibility, keyboard shortcuts, color contrast, and font size options) are clearly explained.

Postconditions:

- User has access to and understands the accessibility features

Expected Result:

- User can access and comprehend the documentation on accessibility features, and can utilize these features accordingly

Severity: Minor

Priority: Low

Type of Testing: Documentation Review, Usability Testing

Test Case Approach: Positive

-----

## **Data - 17**

### **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: SMTP-SMT-001

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

Test Case Description: As a user, I want to ensure that the platform's content moderation algorithms can accurately detect and filter inappropriate content to maintain a safe and positive environment.

Test Suite: Safety and Moderation Tools - Content Moderation

Test Priority: High

Preconditions:

- User is logged in
- User has access to posting features

Test Data: A variety of user-generated posts with both appropriate and inappropriate content

Test Steps:

1. Log in to the platform as a user.
2. Create a new post containing appropriate content.
3. Create a new post containing inappropriate content.
4. Verify that the post with appropriate content is displayed.
5. Verify that the post with inappropriate content is not displayed or is marked as inappropriate.

Postconditions:

- User's posts are saved
- Inappropriate content is filtered

Expected Result: Content moderation algorithms accurately filter inappropriate user-generated content.

Severity: Major

Priority: High

Type of Testing: Functional Testing, System Testing, Acceptance Testing

Test Case Approach: Positive



Test Case ID: SMTP-SMT-002

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

Test Case Description: As a user, I want to ensure that I can easily report harmful or inappropriate content on the platform, so that moderators can review and remove it accordingly.

Test Suite: Safety and Moderation Tools - Reporting Tools

Test Priority: High

Preconditions:

- User is logged in
- User has access to posting features

Test Data: A post with inappropriate content

Test Steps:

1. Log in to the platform as a user.
2. Create a new post containing inappropriate content.
3. Locate the "Report" option for the post.
4. Select "Report" and choose a reason for the report.
5. Verify that the report is submitted successfully.

Postconditions:

- User's report is saved

Expected Result: Users can successfully report harmful or inappropriate content for moderator review.

Severity: Major

Priority: High

Type of Testing: Functional Testing, System Testing, Acceptance Testing

Test Case Approach: Positive

Test Case ID: SMTP-SMT-003

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

Test Case Description: As a user, I want to ensure that I can block other users to prevent them from interacting

with my content or messaging me.

Test Suite: Safety and Moderation Tools - Blocking Capabilities

Test Priority: High

Preconditions:

- User is logged in
- User has access to user profiles

Test Data: A user profile

Test Steps:

1. Log in to the platform as a user.
2. Access a user profile that you wish to block.
3. Locate the "Block" option for the user.
4. Select "Block" and confirm the action.
5. Verify that the blocked user can no longer interact with your content or message you.

Postconditions:

- User is blocked from interacting with your content or messaging you

Expected Result: Blocking capabilities function correctly, preventing interactions between the blocked user and the blocking user.

Severity: Major

Priority: High

Type of Testing: Functional Testing, System Testing, Acceptance Testing

Test Case Approach: Positive

Test Case ID: SMTP-SMT-004

Test Case Title: Verify that comment filters prevent offensive language.

Test Case Description: As a user, I want to ensure that the platform has comment filters in place to prevent the display of offensive language in comments.

Test Suite: Safety and Moderation Tools - Comment Filters

Test Priority: High

Preconditions:

- User is logged in
- User has access to posts with comments

Test Data: Comments containing offensive language

Test Steps:

1. Log in to the platform as a user.
2. Access a post with existing comments.
3. Create a new comment containing offensive language.
4. Verify that the offensive language is either filtered or not displayed.

Postconditions:

- Offensive language is filtered or not displayed

Expected Result: Comment filters effectively prevent the display of offensive language in comments.

Severity: Major

Priority: High

Type of Testing: Functional Testing, System Testing, Acceptance Testing

Test Case Approach: Negative

Test Case ID: SMTP-SMT-005

Test Case Title: Verify that community guidelines are enforced consistently.

Test Case Description: As a user, I want to ensure that the platform's community guidelines are enforced consistently across all users and content.

Test Suite: Safety and Moderation Tools - Community Guidelines Enforcement

Test Priority: High

Preconditions:

- Various user-generated content exists.

Test Data: A set of user-generated posts both adhering to and violating community guidelines

#### Test Steps:

1. Log in to the platform as a moderator.
2. Review a set of user-generated posts.
3. Identify posts that adhere to and violate community guidelines.
4. Verify that posts adhering to community guidelines remain available.
5. Verify that posts violating community guidelines are removed or moderated accordingly.

#### Postconditions:

- Content adhering to community guidelines remains available
- Content violating community guidelines is removed or moderated accordingly

Expected Result: Community guidelines are enforced consistently across all users and content.

Severity: Major

Priority: High

Type of Testing: Functional Testing, System Testing, Acceptance Testing

Test Case Approach: Positive and Negative

These test cases are structured and comprehensive, ensuring thorough coverage of all acceptance criteria, contributing to a high-quality software product by identifying potential defects early in the development cycle.

-----

#### **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.

Test Suite: User Interface Localization

Test Priority: High

Preconditions:

- User is logged in
- User has access to account settings

Test Data: No test data needed

Test Steps:

1. Navigate to account settings
2. Select "Language & Region" from the menu
3. Choose a preferred language from the list
4. Save the changes

Postconditions:

- The platform's interface is displayed in the chosen language
- The language preference is saved and applied consistently throughout the platform

Expected Result: The test case passes if the platform allows users to choose their preferred language for the interface and saves the preference for future use.

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 offers content translation features for posts and messages.

Test Suite: Content Translation

Test Priority: High

Preconditions:

- User is logged in
- User has access to a post or message in a language different from their preferred interface language

Test Data: A post or message in a language different from the user's preferred interface language

Test Steps:

1. Locate a post or message in a different language
2. Click on the "Translate" button

3. Verify that the content is translated into the user's preferred interface language

Postconditions:

- The post or message is displayed in the translated language

Expected Result: The test case passes if the platform translates the content accurately and consistently.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Localization Testing

Test Case Approach: Positive

Test Case ID: TC003

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

Test Case Description: This test case aims to confirm that the platform saves and applies language preferences consistently.

Test Suite: Language Preferences

Test Priority: High

Preconditions:

- User is logged in
- User has chosen their preferred language

Test Data: No test data needed

Test Steps:

1. Log out and log back in
2. Navigate to a different page on the platform
3. Verify the language of the interface

Postconditions:

- The platform's interface is displayed in the user's chosen language

Expected Result: The test case passes if the platform consistently displays the user's preferred language across different pages and sessions.

Severity: Major

Priority: High

Type of Testing: Regression Testing, Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

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

Test Case Description: This test case aims to verify that the social media platform supports a wide range of languages.

Test Suite: Language Support

Test Priority: Medium

Preconditions:

- User is logged in

Test Data: No test data needed

Test Steps:

1. Navigate to the language selection menu
2. Verify the list of supported languages

Expected Result: The test case passes if the platform supports a wide range of languages, covering major world regions and accounting for linguistic diversity.

Severity: Minor

Priority: Medium

Type of Testing: Compatibility Testing, Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that negative scenarios for language selection are handled gracefully

Test Case Description: This test case aims to ensure that the platform handles negative scenarios for language



selection gracefully.

Test Suite: Edge Cases for Language Selection

Test Priority: Low

Preconditions:

- User is logged in

Test Data: No test data needed

Test Steps:

1. Select an unsupported language from the language selection menu
2. Verify if the platform displays a clear message indicating that the language is not supported

Expected Result: The test case passes if the platform handles unsupported language selections gracefully and provides a clear message for the user.

Severity: Trivial

Priority: Low

Type of Testing: Non-Functional Testing, Usability Testing

Test Case Approach: Negative

-----

**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 ID: TC-CPI-001

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

Test Case Description: This test case verifies the ability for users to link their accounts with other social media platforms through the cross-platform integration feature.

Test Suite: User Account Management

Test Priority: High

Preconditions:

- User is logged into the social media platform
- User has access to the account settings page

Test Data: User credentials for the social media platform and the external platform being linked

Test Steps:

1. Navigate to the account settings page
2. Select "Link Accounts" option

3. Choose the external social media platform to link
4. Enter user credentials for the external platform
5. Authorize the connection between the two accounts

Postconditions:

- The external account should be displayed as linked in the user's account settings
- Users should have the option to unlink the account

Expected Result: The user's account is successfully linked with the external platform without any errors or issues.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-CPI-002

Test Case Title: Verify that content can be cross-posted to other platforms

Test Case Description: This test case verifies the ability for users to cross-post content to other social media platforms using the cross-platform integration feature.

Test Suite: Content Sharing

Test Priority: High

Preconditions:

- User is logged into both the social media platform and the external platform
- User has access to the content sharing page

Test Data: A test post, image, or video

Test Steps:

1. Create a new post, attach an image, or upload a video
2. Select the option to cross-post the content
3. Choose the external social media platform to cross-post

#### 4. Confirm the cross-posting action

##### Test Postconditions:

- The content should be successfully shared on both the social media platform and the external platform

Expected Result: The content is successfully cross-posted to the external platform without any errors or issues, and can be viewed by the user's connections on both platforms.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-CPI-003

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

Test Case Description: This test case verifies the ability for users to sign in using single sign-on (SSO) through the cross-platform integration feature.

Test Suite: User Authentication

Test Priority: Medium

##### Preconditions:

- User has an account on the social media platform and an external platform that supports SSO
- The external platform's SSO is enabled for the social media platform

Test Data: User credentials for the external platform

##### Test Steps:

1. Navigate to the login page
2. Select the option to sign in with the external platform
3. Enter user credentials for the external platform

##### Postconditions:

- User is logged into the social media platform without entering their social media credentials

Expected Result: Users can sign in using SSO without any issues or errors, maintaining a seamless user

experience.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-CPI-004

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

Test Case Description: This test case verifies the ability for users to integrate external media sources into their posts through the cross-platform integration feature.

Test Suite: Content Creation

Test Priority: Medium

Preconditions:

- User is logged into the social media platform
- User has access to the content creation page

Test Data: External media source link (e.g., YouTube video, SoundCloud audio, etc.)

Test Steps:

1. Navigate to the content creation page
2. Choose the option to attach external media
3. Enter the link for the external media source
4. Complete the content creation process

Postconditions:

- The external media source should be properly embedded in the post

Expected Result: Users are able to integrate external media sources into their posts without any issues, enhancing the content shareability.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-CPI-005

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

Test Case Description: This test case verifies the overall user experience of cross-platform content sharing through the cross-platform integration feature.

Test Suite: Content Sharing

Test Priority: Low

Preconditions:

- User is logged into both the social media platform and an external platform
- User has access to the content sharing page

Test Data: Test posts, images, and videos

Test Steps:

1. Create a new post, attach an image, or upload a video
2. Select the option to cross-post the content
3. Choose the external platform to cross-post
4. Verify the content on the external platform after cross-posting

Test Postconditions:

- Content should be successfully shared on both the social media platform and the external platform

Expected Result: The user experience of cross-platform content sharing is seamless, with no issues, errors, or discrepancies between the two platforms.

Severity: Major

Priority: Low

Type of Testing: User Experience 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: This test case verifies that registered users can earn points and badges as a reward for their activities on the social media platform.

Test Suite: Gamification and Rewards

Test Priority: High

Preconditions:

- User account is registered and logged in
- User has performed activities such as posting, commenting, or sharing content

Test Data: User accounts with different activity histories

Test Steps:

1. Log in as a registered user.
2. Perform various activities like posting, commenting, or sharing content.
3. Check the user profile for updated points and badges.

Postconditions:

- Points and badges are accurately reflected in the user profile.
- Different activities result in different points and badge awards.

Expected Result: Users can earn points and badges based on their activities, with accurate reflections in their user profiles.

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: This test case verifies that leaderboards display the top users on the social media platform, either ranked by points or achievements.

Test Suite: Gamification and Rewards

Test Priority: High

Preconditions:

- User accounts with varying points and badges are present

Test Steps:

1. Navigate to the leaderboard section.
2. Select the option to view top users by points or badges.
3. Verify that the list displays the correct users in the correct order.

Postconditions:

- Top users by points or badges are accurately displayed in the leaderboard section.

Expected Result: Leaderboards display the top users based on points or achievements, with accurate rankings and user listings.

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: This test case verifies that challenges are available for users to participate in, allowing them to earn rewards upon completion.

Test Suite: Gamification and Rewards

Test Priority: Medium

Preconditions:

- User account is registered and logged in

Test Steps:

1. Log in as a registered user.
2. Navigate to the challenges section.
3. Select a challenge to participate in.
4. Complete the challenge according to the instructions.
5. Check for the reward upon completion of the challenge.

Postconditions:

- Users can participate in challenges.
- Users receive rewards upon completion of challenges.

Expected Result: Challenges are available for users to complete, and rewards are given upon completion.

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: This test case verifies that users can receive various types of rewards for their activities and achievements, such as virtual goods, discounts, or exclusive content.

Test Suite: Gamification and Rewards

Test Priority: Medium

Preconditions:

- User account has earned points or badges through activities or achievements

#### Test Steps:

1. Log in as a registered user.
2. Navigate to the rewards section.
3. Check the available rewards for the user.
4. Ensure the rewards include virtual goods, discounts, or exclusive content.

Expected Result: Users can receive a variety of rewards, including virtual goods, discounts, or exclusive content, based on their earned points and badges.

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: This test case verifies that users have access to a dedicated section where they can view their progress, achievements, and rewards.

Test Suite: Gamification and Rewards

Test Priority: Medium

Preconditions:

- User account is registered and logged in

#### Test Steps:

1. Log in as a registered user.
2. Navigate to the user profile or dedicated gamification section.
3. Check for the availability of progress, achievements, and rewards information.

Expected Result: Users have access to a dedicated section where they can view their progress, achievements,

and rewards.

Severity: Major

Priority: Medium

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 ID: TCT-001

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

Test Case Description: As a quality assurance engineer, I need to verify that users can input a task title during the process of creating a new task.

Test Suite: Task Creation

Test Priority: High

Preconditions:

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

Test Data: "Test task title"

Test Steps:

1. Navigate to the task creation page.
2. Input "Test task title" in the task title field.
3. Click the "Create Task" button.

Postconditions:

- The system saves the task with the specified title.
- The user is redirected to the task details page.

Expected Result: The task is saved with the given title, "Test task title."

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 enter a task description

Test Case Description: As a quality assurance engineer, I need to verify that users can input a task description during the process of creating a new task.

Test Suite: Task Creation

Test Priority: High

Preconditions:

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

Test Data: "This is a test task description."

Test Steps:

1. Navigate to the task creation page.
2. Input "This is a test task description." in the task description field.
3. Click the "Create Task" button.

Postconditions:

- The system saves the task with the specified description.
- The user is redirected to the task details page.

Expected Result: The task is saved with the provided description, "This is a test task description."

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TCT-003

Test Case Title: Verify that users can set a task due date

Test Case Description: As a quality assurance engineer, I need to verify that users can input a task due date during the process of creating a new task.

Test Suite: Task Creation

Test Priority: High

Preconditions:

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

Test Data: 2023-12-31 (a valid future date)

Test Steps:

1. Navigate to the task creation page.
2. Input a valid future date (e.g., 2023-12-31) in the task due date field.
3. Click the "Create Task" button.

Postconditions:

- The system saves the task with the specified due date.
- The user is redirected to the task details page.

Expected Result: The task is saved with the given due date, 2023-12-31.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TCT-004

Test Case Title: Verify that users can set a task priority level

Test Case Description: As a quality assurance engineer, I need to verify that users can input a task priority level during the process of creating a new task.

Test Suite: Task Creation

Test Priority: High

Preconditions:

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

Test Data: High

Test Steps:

1. Navigate to the task creation page.
2. Select "High" from the task priority dropdown list.
3. Click the "Create Task" button.

Postconditions:

- The system saves the task with the specified priority level.
- The user is redirected to the task details page.

Expected Result: The task is saved with the selected priority level, High.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TCT-005

Test Case Title: Verify that users cannot create a task without entering a title

Test Case Description: As a quality assurance engineer, I need to verify that users cannot create a task without providing a task title.

Test Suite: Task Creation

Test Priority: High



Preconditions:

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

Test Data: No test data needed

Test Steps:

1. Navigate to the task creation page.
2. Leave the task title field empty.
3. Click the "Create Task" button.

Postconditions:

- The system displays an error message for the missing task title.

Expected Result: The system requires task title input and displays an error message if left empty.

Severity: Major

Priority: High

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 ID: TCT-001

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

Test Case Description: As a user, verify that you can select a team member from the available list to assign a task.

Test Suite: Task Assignment

Test Priority: High

Preconditions:

- User is logged in
- A new task has been created

Test Data: A list of available team members

Test Steps:

1. Create a new task
2. Enter task details, including a title, description, and deadline
3. Click on the "Assign to" field

4. Select a team member from the dropdown list

5. Save the task

Postconditions:

- The selected team member is displayed as the task assignee

Expected Result:

- The user can successfully select a team member from the available list to assign the task.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TCT-002

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

Test Case Description: As a user, verify that assignees receive notifications via email or in-app messaging when they are assigned a task.

Test Suite: Notification System

Test Priority: Medium

Preconditions:

- User is logged in
- A new task has been created
- A team member has been assigned to the task

Test Data: A newly assigned task

Test Steps:

1. Log in as the assigned team member
2. Check for a notification about the new task in the app and/or via email

Postconditions:

- A notification is displayed

Expected Result:

- The assigned team member receives a notification about the new task.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TCT-003

Test Case Title: Verify that assigned tasks are displayed in the assignee's task list

Test Case Description: As an assignee, verify that assigned tasks are displayed in the task list, allowing easy tracking and management.

Test Suite: Task List Display

Test Priority: High

Preconditions:

- User is logged in
- A new task has been created
- A team member has been assigned to the task

Test Data: A newly assigned task

Test Steps:

1. Log in as the assigned team member
2. Navigate to the task list
3. Verify that the newly assigned task is displayed

Postconditions:

- The assigned task is visible

Expected Result:

- Assigned tasks are displayed in the assignee's task list.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TCT-004

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

Test Case Description: As a user, verify that existing tasks can be reassigned to different team members, updating the task assignee and notifying the new assignee.

Test Suite: Task Reassignment

Test Priority: High

Preconditions:

- User is logged in
- A task has been created and assigned to a team member

Test Data: A different team member

Test Steps:

1. Navigate to the task list
2. Select the task assigned to the original team member
3. Click on the "Reassign" button
4. Select the new team member from the dropdown list
5. Save the reassigned task

Postconditions:

- The selected team member is displayed as the new task assignee
- The original assignee receives a notification about the reassignment

Expected Result:

- Users can successfully reassign tasks to different team members.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TCT-005

Test Case Title: Verify that task assignment history is tracked

Test Case Description: As a user, verify that the task assignment history is tracked and accessible, providing a clear record of responsibility and progress.

Test Suite: Task History

Test Priority: Medium

Preconditions:

- User is logged in
- A task has been created and assigned to at least one team member
- The task has been reassigned at least once

Test Data: A task with multiple assignees

Test Steps:

1. Navigate to a task
2. Access the task history
3. Verify the list of assignees and assignment dates

Postconditions:

- The task history displays the list of assignees and assignment dates

Expected Result:

- The task assignment history is tracked and accessible.

Severity: Major

Priority: Medium

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 ID: TCTM-001

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

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

Test Suite: Task Reminder Settings

Test Priority: High

Preconditions:

- User is logged in
- A task with a deadline has been created

Test Data: A task with a deadline set for the next day

Test Steps:

1. Navigate to the task details page
2. Click on the "Set Reminder" button
3. Select a time for the reminder (e.g., 1 hour before the deadline)
4. Save the reminder settings
5. Wait for the specified time

Postconditions:

- The reminder notification is displayed

Expected Result: The user receives a reminder notification for the task before the deadline, as set.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive



Test Case ID: TCTM-002

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

Test Case Description: As a user, I want to set different urgency levels for my reminders, so I can prioritize my tasks accordingly.

Test Suite: Task Reminder Settings

Test Priority: High

Preconditions:

- User is logged in
- A task with a deadline has been created

Test Data: A task with a deadline set for tomorrow, with three different urgency levels (low, medium, high)

Test Steps:

1. Navigate to the task details page
2. Click on the "Set Reminder" button
3. Select an urgency level (low, medium, high)
4. Save the reminder settings
5. Wait for the specified time based on the urgency level

Postconditions:

- The reminder notification is displayed, taking into account the selected urgency level

Expected Result: The user receives a reminder notification for the task before the deadline, with the correct urgency level.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCTM-003

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

Test Case Description: As a user, I want to receive notifications from the system for any upcoming task deadlines, so I can manage my tasks effectively.

Test Suite: Task Deadline Notifications

Test Priority: High

Preconditions:

- User is logged in
- A task with a deadline has been created
- The deadline is set for the next day

Test Data: N/A

Test Steps:

1. Log in as a user
2. Ensure that no reminders have been set for the task
3. Wait for the system-defined time before the deadline

Postconditions:

- The deadline notification is displayed

Expected Result: The user receives a notification for the task's upcoming deadline, even if no reminder was set.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCTM-004

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

Test Case Description: As a user, I want to be able to select my preferred reminder delivery method, so I can stay informed in the way that works best for me.

Test Suite: Task Reminder Settings

Test Priority: Medium

Preconditions:

- User is logged in
- A task with a deadline has been created

Test Data: N/A (User's preferred delivery method is set to email)

Test Steps:

1. Navigate to the task details page
2. Click on the "Set Reminder" button
3. Set a reminder for the task
4. Change the delivery method to email
5. Save the reminder settings

Postconditions:

- The reminder settings are saved with the selected delivery method

Expected Result: The user receives the reminder for the task via email, as set.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCTM-005

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

Test Case Description: As a user, I want to have the ability to adjust or disable my reminders, so I can manage my notifications and stay focused on my tasks.

Test Suite: Task Reminder Settings

Test Priority: Medium

Preconditions:

- User is logged in
- A task with a reminder has been created

Test Data: N/A

Test Steps:

1. Navigate to the task details page
2. Click on the "Set Reminder" button
3. Adjust the reminder settings (time, urgency level, or delivery method) or disable the reminder
4. Save the changes

Postconditions:

- The reminder settings are saved with the adjusted or disabled reminder

Expected Result: The user can adjust or disable reminders for the task, and the changes are saved correctly.

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: TCT-TP001

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

Test Case Description: This test case verifies that a registered user is able to assign priority levels (High, Medium, Low) to newly created tasks.

Test Suite: Task Prioritization

Test Priority: High

Preconditions:

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

Test Data: New task information (title, description, deadline, etc.)

Test Steps:

1. Navigate to the task creation page

2. Enter new task information
3. Select a priority level for the task (High, Medium, Low)
4. Save or create the new task

Postconditions:

- The new task is saved in the system
- The new task displays the selected priority level

Expected Result: The new task has the selected priority level (High, Medium, Low) assigned.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TCT-TP002

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

Test Case Description: This test case verifies that tasks are displayed and sorted according to their priority levels (High, Medium, Low).

Test Suite: Task Prioritization

Test Priority: Medium

Preconditions:

- User is registered and logged in
- User has created tasks with different priority levels

Test Data: N/A

Test Steps:

1. Navigate to the task list or dashboard page
2. Verify that tasks are sorted by priority (High, Medium, Low)

Postconditions:

- Tasks are displayed and sorted according to their priority levels

Expected Result: Tasks with higher priority levels (High) are displayed before tasks with lower priority levels (Medium, Low).

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TCT-TP003

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

Test Case Description: This test case verifies that a registered user can modify the priority level of an existing task.

Test Suite: Task Prioritization

Test Priority: Medium

Preconditions:

- User is registered and logged in
- User has created a task with a specific priority level

Test Data: New priority level for the existing task

Test Steps:

1. Navigate to the task details page
2. Change the priority level of the existing task
3. Save the changes

Postconditions:

- The task priority level has been updated

Expected Result: The task displays the new priority level (High, Medium, Low) on the task details page and task list.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TCT-TP004

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

Test Case Description: This test case verifies that tasks are visually distinguishable based on their priority levels (High, Medium, Low).

Test Suite: Task Prioritization

Test Priority: Medium

Preconditions:

- User is registered and logged in
- User has created tasks with different priority levels

Test Data: N/A

Test Steps:

1. Navigate to the task list or dashboard page
2. Verify that tasks with different priority levels are visually distinguishable (e.g., color, icons, symbols)

Postconditions:

- Tasks with different priority levels are displayed with distinguishable visual elements

Expected Result: High-priority tasks have clearly identifiable visual elements that separate them from medium- and low-priority tasks.

Severity: Major



Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TCT-TP005

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

Test Case Description: This test case verifies that tasks without a manually assigned priority level default to the Medium priority level.

Test Suite: Task Prioritization

Test Priority: Low

Preconditions:

- User is registered and logged in
- User has created a task without manually assigning a priority level

Test Data: N/A

Test Steps:

1. Navigate to the task details page
2. Verify that the task has a default priority level (Medium)

Postconditions:

- The task has a default priority level (Medium)

Expected Result: Tasks with no priority level assigned default to the Medium priority level.

Severity: Minor

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 ID: TCT-001

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

Test Case Description: As a user, I want to create a subtask so that I can break down complex tasks into manageable parts. This test case verifies if the system allows users to create subtasks associated with a main

task.

Test Suite: Subtask Management

Test Priority: High

Preconditions:

- User must be logged in.
- User must have access to create tasks and subtasks.
- A main task must already exist.

Test Data: Main task ID

Test Steps:

1. Navigate to the main task page.
2. Click on the "Add Subtask" button.
3. Enter the subtask details, including a name, description, and due date.
4. Assign the subtask to a team member.
5. Save the subtask.

Postconditions:

- The new subtask is visible under the main task.
- The subtask appears in the list of all subtasks for the user.

Expected Result:

- The system should create and display the subtask under the main task.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-002

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

Test Case Description: As a user, I want subtasks to have their own due dates and assignees so that I can

manage tasks and projects more efficiently. This test case verifies if the system allows users to set due dates and assignees for subtasks independently.

Test Suite: Subtask Management

Test Priority: High

Preconditions:

- User must be logged in.
- User must have access to create tasks and subtasks.
- A main task must already exist.

Test Data: Main task ID, assignee name, subtask due date

Test Steps:

1. Navigate to the main task page.
2. Create a new subtask with a specific due date and assign it to a team member.
3. Save the subtask.

Postconditions:

- The subtask displays the assigned team member.
- The subtask displays the specified due date.

Expected Result:

- The system should display the assigned team member and due date for the subtask.

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 tracks the progress of each subtask

Test Case Description: As a user, I want the system to track the progress of each subtask so that I can monitor task completion. This test case verifies if the system correctly tracks subtask completion and updates the main

task's progress accordingly.

Test Suite: Subtask Management

Test Priority: High

Preconditions:

- User must be logged in.
- User must have access to create tasks and subtasks.
- A main task with subtasks must already exist.

Test Data: Subtask ID, subtask completion status

Test Steps:

1. Navigate to the main task page.
2. Change the completion status of a subtask.
3. Save the main task.

Postconditions:

- The subtask displays the new completion status.
- The main task's progress percentage should update based on the subtask's completion status.

Expected Result:

- The system should display the accurate completion status of the subtask and update the main task's progress.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-004

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

Test Case Description: As a user, I want subtasks to be marked as complete independently. This test case verifies if the system allows subtasks to be marked as complete even if the main task is incomplete.

Test Suite: Subtask Management

Test Priority: High

Preconditions:

- User must be logged in.
- User must have access to create tasks and subtasks.
- A main task with subtasks must already exist.

Test Data: Subtask ID, subtask completion status

Test Steps:

1. Navigate to the main task page.
2. Change the completion status of a subtask.
3. Save the subtask.

Postconditions:

- The subtask displays the new completion status.
- The main task's completion status remains unchanged.

Expected Result:

- The system should display the subtask's completion status accurately and independently from the main task.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-005

Test Case Title: Verify that dependencies between subtasks can be set

Test Case Description: As a user, I want dependencies between subtasks so that I can manage task priorities and clarity. This test case verifies if the system allows users to set dependencies between subtasks, so their completion order is maintained.

Test Suite: Subtask Management

Test Priority: High

Preconditions:

- User must be logged in.
- User must have access to create tasks and subtasks.
- A main task with subtasks must already exist.

Test Data: Subtask IDs and dependency relationship

Test Steps:

1. Navigate to the main task page.
2. Set a dependency relationship between two subtasks.
3. Save the main task.

Postconditions:

- The subtasks display their dependency relationship.
- The order of subtask completion is maintained according to the dependency relationship.

Expected Result:

- The system should display the subtasks with the correct dependency relationship and enforce the completion order accordingly.

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-001

Test Case Title: Verify that users can set dependencies between tasks

Test Case Description: As a Quality Assurance Engineer, I need to verify that the Task Management Tool allows users to define dependencies between tasks so that tasks are completed in the correct sequence.

Test Suite: Task Dependencies

Test Priority: High

Preconditions:

- User is logged in
- User has access to create tasks
- User has at least two tasks created

Test Data: Two existing tasks, Task A and Task B, where Task B depends on Task A



#### Test Steps:

1. Navigate to the task list or project board.
2. Select an existing task (Task A) and open the task details.
3. Locate the "Dependencies" section and click on "Add Dependency".
4. Select the second existing task (Task B) from the task list.
5. Save the changes.

#### Postconditions:

- Task B is displayed as a dependent task in Task A's details
- Task A's completion status affects Task B's completion status

Expected Result: After setting the dependency, Task B should be displayed in Task A's dependencies list.

Task B's completion status should be blocked by Task A's completion status, i.e., Task B cannot be marked as complete if Task A is not completed.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TCT-002

Test Case Title: Verify that the system displays dependencies in the task details

Test Case Description: As a Quality Assurance Engineer, I need to verify that the Task Management Tool displays the dependencies in the task details so that users can understand the order in which tasks should be completed.

Test Suite: Task Dependencies

Test Priority: Medium

Preconditions:

- User is logged in
- User has access to tasks with dependencies

Test Data: Task A, which has Task B as a dependent task

Test Steps:

1. Navigate to the task list or project board.
2. Select Task A and open the task details.
3. Locate the "Dependencies" section.

Postconditions:

- Task B is displayed in the dependencies list

Expected Result: Task B should be displayed in Task A's dependencies list in the task details.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TCT-003

Test Case Title: Verify that tasks cannot be marked as complete if dependent tasks are not completed

Test Case Description: As a Quality Assurance Engineer, I need to verify that the Task Management Tool prevents users from marking tasks as complete when their dependent tasks are not completed to ensure tasks are completed in sequence.

Test Suite: Task Dependencies

Test Priority: High

Preconditions:

- User is logged in
- User has access to tasks with dependencies

- Task A is a dependent task for Task B
- Task B is incomplete

Test Data: Task A and Task B, where Task B depends on Task A

Test Steps:

1. Navigate to the task list or project board.
2. Select Task A and attempt to mark it as complete.

Postconditions:

- Task A should not be marked as complete

Expected Result: Task A cannot be marked as complete while Task B is incomplete.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Negative

---

Test Case ID: TCT-004

Test Case Title: Verify that users can update or remove dependencies as needed

Test Case Description: As a Quality Assurance Engineer, I need to verify that the Task Management Tool allows users to update or remove task dependencies to modify the task sequence as required.

Test Suite: Task Dependencies

Test Priority: Medium

Preconditions:

- User is logged in
- User has access to tasks with dependencies

Test Data: Task A, which has Task B as a dependent task

Test Steps:

1. Navigate to the task list or project board.
2. Select Task A and open the task details.
3. Locate the "Dependencies" section.
4. Remove Task B from the dependencies list.
5. Save the changes.

Postconditions:

- Task B is no longer displayed in the dependencies list

Expected Result: Task B should be removed from Task A's dependencies list in the task details. Task B's completion status should no longer be blocked by Task A's completion status.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TCT-005

Test Case Title: Verify that the system provides alerts or warnings if a dependent task is not completed

Test Case Description: As a Quality Assurance Engineer, I need to verify that the Task Management Tool alerts or warns users if a dependent task is not...

-----

**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: As a user, I want to add a comment to a task to provide my input or feedback.

Test Suite: Task Commenting

Test Priority: High

Preconditions:

\* User must be logged in

\* A task must exist

Test Data: A comment text string

Test Steps:

1. Navigate to the specific task page
2. Locate and click on the "Add Comment" button
3. Enter the comment text in the input field
4. Click the "Post Comment" button

Postconditions:

- \* The comment should be displayed on the task page
- \* The user who posted the comment should see it as their most recent comment

Expected Result: The user can successfully add a new comment to the task.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that comments can include attachments

Test Case Description: As a user, I want to attach a file to my comment to share relevant information or resources.

Test Suite: Task Commenting

Test Priority: Medium

Preconditions:

- \* User must be logged in
- \* A task must exist

Test Data: A file to attach to the comment

Test Steps:

1. Navigate to the specific task page
2. Locate and click on the "Add Comment" button
3. Enter the comment text in the input field

4. Click the "Attach File" button and select the required file

5. Click the "Post Comment" button

Postconditions:

- \* The comment should be displayed on the task page with the attached file

Expected Result: The user can successfully attach a file to a comment on the task.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users can reply to comments

Test Case Description: As a user, I want to reply to a comment to engage in a conversation or discussion about the task.

Test Suite: Task Commenting

Test Priority: Medium

Preconditions:

- \* User must be logged in

- \* A task must exist with at least one comment

Test Data: A reply text string

Test Steps:

1. Navigate to the specific task page

2. Locate an existing comment and click on the "Reply" button below it

3. Enter the reply text in the input field

4. Click the "Post Reply" button

Postconditions:

- \* The reply should be displayed below the original comment

Expected Result: The user can successfully reply to an existing comment on the task.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that users are notified of new comments

Test Case Description: As a user, I want to receive notifications when there are new comments on tasks I'm involved in to stay updated and engaged.

Test Suite: Task Notifications

Test Priority: High

Preconditions:

- \* User must be logged in
- \* A task must exist with at least one other user assigned
- \* The user must not be currently viewing the task page

Test Data: A new comment added to the task

Test Steps:

1. Log in as the secondary user assigned to the task
2. Add a new comment to the task
3. Wait for the notification to be triggered (email, in-app, etc.)

Postconditions:

- \* The primary user should receive a notification of the new comment

Expected Result: The user is notified of new comments on tasks they are involved in.

Severity: Critical

Priority: High

Type of Testing: Functional Testing



Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that comments are displayed in chronological order

Test Case Description: As a user, I want to see comments in the correct order based on their timestamp to follow the conversation and maintain context.

Test Suite: Task Commenting

Test Priority: High

Preconditions:

- \* User must be logged in
- \* A task must exist with at least three comments

Test Data: No test data needed

Test Steps:

1. Navigate to the specific task page
2. Scroll through the comments section

Expected Result: Comments should be displayed in reverse chronological order, with the most recent comment at the top.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

-----

**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: TC-FAT-001

Test Case Title: Verify that users can attach multiple files to a task

Test Case Description: Validate the system's ability to allow users to attach multiple files to a task, ensuring proper functionality and file management.

Test Suite: File Attachments

Test Priority: High

Preconditions:

- User is logged in.
- A new or existing task is open.

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 local system.
4. Click "Open" to upload the file.
5. Repeat steps 2 and 4 for the remaining two files.
6. Verify that the three files are displayed in the attachments list.

Postconditions:

- Files are visible in the attachments list.

Expected Result:

- Users can successfully attach multiple files to a task.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-FAT-002

Test Case Title: Verify that attachments can include various file types (documents, images, etc.)

Test Case Description: Ensure that the system supports various file types, allowing users to attach and manage files effectively.

Test Suite: File Attachments

Test Priority: Medium

Preconditions:

- User is logged in.
- A new or existing task is open.

Test Data: Five different file types (e.g., .docx, .jpg, .pdf, .xlsx, .pptx)

#### Test Steps:

1. Navigate to the task details page.
2. Click on the "Attach File" button.
3. Select the first file from the local system.
4. Click "Open" to upload the file.
5. Repeat steps 2 to 4 for the remaining four files of different types.
6. Verify that all five files of various types are displayed in the attachments list.

#### Postconditions:

- Files of various types are visible in the attachments list.

#### Expected Result:

- Users can attach different file types to a task.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-FAT-003

Test Case Title: Verify that users can view and download attachments

Test Case Description: Validate the system's ability to display and download attachments, ensuring proper access and retrieval of files.

Test Suite: File Attachments

Test Priority: High

#### Preconditions:

- User is logged in.
- A task with at least one attachment exists.

Test Data: No test data needed

#### Test Steps:

1. Navigate to a task with an attachment.
2. Locate the attachment in the attachments list.
3. Click on the attachment to view it.
4. Verify that the attachment opens in the appropriate application or web viewer.
5. Click the "Download" button to download the attachment.
6. Verify that the downloaded file matches the original.

Postconditions:

- Attachments can be viewed and downloaded.

Expected Result:

- Files can be successfully viewed and downloaded from the attachments list.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-FAT-004

Test Case Title: Verify that the system limits the size of attachments to prevent data overload

Test Case Description: Test the system's attachment size limit, ensuring proper resource management and performance.

Test Suite: File Attachments

Test Priority: Medium

Preconditions:

- User is logged in.
- A new or existing task is open.

Test Data: A large file (e.g., >10MB)

Test Steps:

1. Navigate to the task details page.

2. Click on the "Attach File" button.
3. Select the large file from the local system.
4. Click "Open" to attempt uploading the file.
5. Verify that an appropriate error message is displayed, indicating that the file size exceeds the limit.

Postconditions:

- The system prevents uploading files larger than the specified limit.

Expected Result:

- The system restricts the size of attachments based on the configured limit.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Negative

Test Case ID: TC-FAT-005

Test Case Title: Verify that users can remove attachments if needed

Test Case Description: Validate the system's ability to allow users to remove attachments if they are no longer needed.

Test Suite: File Attachments

Test Priority: Low

Preconditions:

- User is logged in.
- A task with at least one attachment exists.

Test Data: No test data needed

Test Steps:

1. Navigate to a task with an attachment.
2. Locate the attachment in the attachments list.
3. Click the "Delete" or "Remove" button associated with the attachment.

4. Verify that a confirmation message is displayed.
5. Confirm the deletion of the attachment.
6. Verify that the attachment no longer appears in the attachments list.

Postconditions:

- Attachments can be removed from a task.

Expected Result:

- Users can successfully remove attachments from a task.

Severity: Major

Priority: Low

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: TCT-RT-001

Test Case Title: Verify that users can set tasks to recur at specified intervals (daily, weekly, monthly)

Test Case Description: As a user, I want to create a recurring task with different frequency options to ensure the system correctly generates instances of the task at the specified intervals.

Test Suite: Recurring Tasks

Test Priority: High

Preconditions:

- User is logged in
- User has access to create tasks

Test Data:

- Task title: "Weekly Report"
- Recurrence pattern: Weekly

Test Steps:

1. Log in as a user.
2. Navigate to the task creation page.
3. Enter the task title "Weekly Report".
4. Select the recurrence pattern "Weekly".
5. Save the task.



Postconditions:

- The system generates a weekly instance of the "Weekly Report" task.

Expected Result:

- The system correctly generates a weekly instance of the "Weekly Report" task.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-RT-002

Test Case Title: Verify that the system generates instances of recurring tasks automatically

Test Case Description: As a user, I want the system to automatically create instances of my recurring tasks so that I do not have to manually create them for each occurrence.

Test Suite: Recurring Tasks

Test Priority: High

Preconditions:

- User is logged in
- User has access to create tasks
- A recurring task exists

Test Data: N/A

Test Steps:

1. Log in as a user.
2. Navigate to the task list page.
3. Verify that instances of the recurring task are automatically generated.

Postconditions:

- Instances of the recurring task are present in the task list.

Expected Result:

- The system automatically generates instances of the recurring task.

Severity: Major

Priority: High

Type of Testing: System Testing

Test Case Approach: Positive

Test Case ID: TCT-RT-003

Test Case Title: Verify that users can modify the recurrence pattern or cancel the recurrence of a task

Test Case Description: As a user, I want to be able to modify the recurrence pattern or cancel the recurrence of a task in case my schedule changes.

Test Suite: Recurring Tasks

Test Priority: Medium

Preconditions:

- User is logged in
- User has access to create tasks
- A recurring task exists

Test Data:

- Task title: "Daily Check-in"
- Recurrence pattern: Daily

Test Steps:

1. Log in as a user.
2. Navigate to the task list page.
3. Locate the "Daily Check-in" task.
4. Edit the recurrence pattern to "Weekly".
5. Save the changes.

Test Steps (For cancel recurrence):

1. Log in as a user.

2. Navigate to the task list page.
3. Locate the "Daily Check-in" task.
4. Cancel the recurrence.
5. Save the changes.

Postconditions:

- Recurrence pattern is updated to weekly
- Recurrence is canceled

Expected Result:

- The recurrence pattern is updated to weekly.
- The recurrence of the task is canceled.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-RT-004

Test Case Title: Verify that the system notifies users of upcoming instances of recurring tasks

Test Case Description: As a user, I want to receive notifications about upcoming instances of my recurring tasks so that I can plan my time accordingly.

Test Suite: Recurring Tasks

Test Priority: Medium

Preconditions:

- User is logged in
- User has access to create tasks
- A recurring task exists with a notification setting

Test Data: N/A

Test Steps:

1. Log in as a user.
2. Ensure notification settings are enabled for the recurring task.
3. Wait for the specified time before the next instance of the recurring task.

Postconditions:

- User receives a notification about the upcoming instance of the task.

Expected Result:

- The user receives a notification about the upcoming instance of the recurring task.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-RT-005

Test Case Title: Verify that users can view a list of all instances of a recurring task

Test Case Description: As a user, I want to see all instances of my recurring tasks so that I can manage them effectively.

Test Suite: Recurring Tasks

Test Priority: Medium

Preconditions:

- User is logged in
- User has access to create tasks
- A recurring task exists

Test Data: N/A

Test Steps:

1. Log in as a user.
2. Navigate to the task list page.
3. Locate the recurring task.

4. Click on the recurring task to view its instances.

Postconditions:

- All instances of the recurring task are displayed.

Expected Result:

- All instances of the recurring task are displayed.

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-TC-001

Test Case Title: Verify that users can add tags and labels to tasks

Test Case Description: As a user, I want to add custom tags and labels to a newly created task to categorize it accordingly.

Test Suite: Task Management

Test Priority: High

Preconditions:

- User is logged in
- User has the necessary permissions to create tasks
- User is on the task creation page

Test Data: "Design", "High Priority"

Test Steps:

1. Navigate to the task creation page.
2. Enter a task name, description, and deadline.
3. Input "Design" as a tag and "High Priority" as a label.
4. Create the task.

Postconditions:

- The task is saved with the "Design" tag and "High Priority" label

Expected Result: Task is created with the custom tags and labels.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TMT-TC-002

Test Case Title: Verify that tags can be customized and edited

Test Case Description: As a user, I want to edit existing tags to maintain consistency and keep the task management tool organized.

Test Suite: Task Management

Test Priority: Medium

Preconditions:

- User is logged in
- User has the necessary permissions to edit tasks
- User has existing tasks with tags

Test Data: "Design" -> "Design Concept"

Test Steps:

1. Navigate to the task list.
2. Select a task with the "Design" tag.
3. Edit the "Design" tag to "Design Concept".
4. Save the changes.

Postconditions:

- The task's tag is updated from "Design" to "Design Concept".

Expected Result: Tag has been successfully edited.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TMT-TC-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 filter and sort tasks based on tags and labels to easily find and manage tasks.

Test Suite: Task Management

Test Priority: High

Preconditions:

- User is logged in
- User has the necessary permissions to view tasks
- User has tasks with various tags and labels

Test Data: "High Priority", "Design Concept", "In Progress"

Test Steps:

1. Navigate to the task list.
2. Select the filter option.
3. Choose "High Priority", "Design Concept", and "In Progress" as the filters.
4. Apply the filters.

Postconditions:

- The task list updates to show tasks matching the selected filters.

Expected Result: Tasks can be filtered and sorted based on tags and labels.

Severity: Major

Priority: High



Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TMT-TC-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 quickly identify tasks using tags and labels to easily manage and prioritize tasks.

Test Suite: Task Management

Test Priority: Medium

Preconditions:

- User is logged in
- User has the necessary permissions to view tasks
- User has tasks with various tags and labels

Test Steps:

1. Navigate to the task list.
2. Verify that the tasks have their respective tags and labels visible.

Postconditions:

- Tags and labels are visible and easily identifiable.

Expected Result: Tags and labels are prominently displayed in the task list.

Severity: Major

Priority: Medium

Type of Testing: Usability Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TMT-TC-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 and labels from tasks to reorganize and re-categorize tasks efficiently.

Test Suite: Task Management

Test Priority: High

Preconditions:

- User is logged in
- User has the necessary permissions to edit tasks
- User has tasks with labels and tags

Test Data: "High Priority", "Design Concept"

Test Steps:

1. Navigate to the task list.
2. Select a task with the "High Priority" and "Design Concept" labels.
3. Remove the "High Priority" and "Design Concept" labels.
4. Save the changes.

Postconditions:

- The task no longer has the "High Priority" and "Design Concept" labels.

Expected Result: Tags and labels can be removed from tasks.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

-----

**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 ID: TCT-001

Test Case Title: Verify that users can update the status of tasks

Test Case Description: This test case aims to verify that users can successfully update the status of tasks from "To Do" to "In Progress" and then to "Completed."

Test Suite: Task Management - Status Updating

Test Priority: High

Preconditions:

- User is logged in
- User has access to a task in the "To Do" status

Test Data: No test data needed

Test Steps:

1. Log in to the Task Management Tool as a user.
2. Navigate to the task list and find a task in the "To Do" status.
3. Click on the task to open it.
4. In the task details, click the "Update Status" button.
5. Select "In Progress" from the status dropdown menu.
6. Click "Save" to update the task status.
7. Repeat steps 4 through 6 to update the task status to "Completed."

Postconditions:

- The task status has been successfully updated in the task list
- The task status is displayed prominently in the task list
- The system provides a notification for the status change

Expected Result:

- The task status updates from "To Do" to "In Progress" and then to "Completed" without any issues
- The task status is correctly displayed in the task list
- The system provides a notification for each status change

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TCT-002

Test Case Title: Verify that the system displays task status prominently in the task list

Test Case Description: This test case aims to verify that task status is displayed prominently in the task list and easy to identify.

Test Suite: Task Management - Status Display

Test Priority: High

Preconditions:

- User is logged in
- User has access to tasks with various statuses (e.g. "To Do", "In Progress", "Completed")

Test Data: No test data needed

Test Steps:

1. Log in to the Task Management Tool as a user.
2. Navigate to the task list.
3. Verify that tasks with different statuses are easily identifiable based on their color-coding and/or icons.
4. Verify that the task status is displayed alongside the task name.

Postconditions:

- The task status is prominently displayed in the task list

Expected Result:

- The task status is displayed in a clear and easily identifiable manner in the task list

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TCT-003

Test Case Title: Verify that users can view the progress percentage of tasks

Test Case Description: This test case aims to verify that users can view the progress percentage of tasks and that the progress percentage is calculated correctly.

Test Suite: Task Management - Progress Percentage

Test Priority: Medium

Preconditions:

- User is logged in
- User has access to tasks with subtasks or checklists
- At least one subtask or checklist item is marked as complete

Test Data: No test data needed

Test Steps:

1. Log in to the Task Management Tool as a user.
2. Navigate to the task list.

3. Find a task with subtasks or checklists and open it.
4. Verify that the progress percentage is displayed prominently in the task details.
5. Verify that the progress percentage is calculated based on the number of completed subtasks or checklist items.

Postconditions:

- The progress percentage is displayed correctly in the task details.

Expected Result:

- The progress percentage is calculated correctly based on the number of completed subtasks or checklist items.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TCT-004

Test Case Title: Verify that task status can be filtered and sorted

Test Case Description: This test case aims to verify that task status can be filtered and sorted in the task list.

Test Suite: Task Management - Status Filtering and Sorting

Test Priority: Medium

Preconditions:

- User is logged in

- User has access to tasks with various statuses (e.g. "To Do", "In Progress", "Completed")

Test Data: No test data needed

Test Steps:

1. Log in to the Task Management Tool as a user.
2. Navigate to the task list.
3. Verify that the task status filter is available.
4. Select a specific task status (e.g. "In Progress") from the filter dropdown menu.
5. Verify that only tasks with the selected status are displayed in the task list.
6. Verify that the task status filter can be cleared.
7. Verify that the task list can be sorted by task status.

Postconditions:

- The task status filter and sorting options are available and functional.

Expected Result:

- The task status filter and sorting options work as expected.

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 provides notifications for status changes

Test Case Description: This test case aims to verify that the system provides notifications for task status changes.

Test Suite: Task Management - Status Notifications

Test Priority: Low

Preconditions:

- User is logged in
- User has access to tasks with various statuses (e.g. "To Do", "In Progress", "Completed")
- Notifications are enabled

Test Data: No test data needed

Test Steps:

1. Log in to the Task Management Tool as a user.
2. Navigate to the task list.
3. Find a task in the "To Do" status and open it.
4. Update the task status to "In Progress" and save the changes.
5. Verify that a notification for the status change is displayed.

Postconditions:

- The system provides a notification for the status change.

Expected Result:

- The system provides notifications for task status changes.

Severity: Major

Priority: Low

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: TCT-TFS-001

Test Case Title: Verify that users can filter tasks based on due date

Test Case Description: This test case verifies that users can apply a due date filter to the task list to view tasks with specific due dates or date ranges.

Test Suite: Task Filtering and Sorting

Test Priority: High

Preconditions:

- User is logged in
- Tasks with various due dates exist

Test Data: No test data needed

Test Steps:

1. Navigate to the task list page
2. Click on the "Filter" button
3. Select "Due Date" in the filter options
4. Choose a specific due date or date range
5. Apply the filter

Postconditions:

- Task list updates to display only tasks within the selected due date or date range

Expected Result:

- The task list displays tasks with the selected due date or date range, indicating that the due date filter works correctly

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-TFS-002

Test Case Title: Verify that users can filter tasks based on priority

Test Case Description: This test case verifies that users can apply a priority filter to the task list to view tasks with specific priority levels.

Test Suite: Task Filtering and Sorting

Test Priority: High

Preconditions:

- User is logged in
- Tasks with various priority levels exist

Test Data: No test data needed

Test Steps:

1. Navigate to the task list page
2. Click on the "Filter" button
3. Select "Priority" in the filter options
4. Choose a specific priority level (e.g., low, medium, high)
5. Apply the filter

Postconditions:

- Task list updates to display only tasks with the selected priority level

Expected Result:

- The task list displays tasks with the selected priority level, indicating that the priority filter works correctly

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-TFS-003

Test Case Title: Verify that users can filter tasks based on assignee

Test Case Description: This test case verifies that users can apply an assignee filter to the task list to view

tasks assigned to specific users.

Test Suite: Task Filtering and Sorting

Test Priority: Medium

Preconditions:

- User is logged in
- Multiple users and tasks exist
- Some tasks are assigned to specific users

Test Data: No test data needed

Test Steps:

1. Navigate to the task list page
2. Click on the "Filter" button
3. Select "Assignee" in the filter options
4. Choose a specific user from the list of assignees
5. Apply the filter

Postconditions:

- Task list updates to display only tasks assigned to the selected user

Expected Result:

- The task list displays tasks assigned to the selected user, indicating that the assignee filter works correctly

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-TFS-004

Test Case Title: Verify that users can sort tasks by due date

Test Case Description: This test case verifies that users can sort the task list by due date to view tasks organized by their due dates.

Test Suite: Task Filtering and Sorting

Test Priority: Medium

Preconditions:

- User is logged in
- Tasks with various due dates exist

Test Data: No test data needed

Test Steps:

1. Navigate to the task list page
2. Click on the "Sort" button
3. Select "Due Date" in the sort options
4. Choose "Ascending" or "Descending" order
5. Apply the sort

Postconditions:

- Task list updates to display tasks organized by their due dates in the selected order

Expected Result:

- The task list displays tasks organized by their due dates in the selected order, indicating that the sort by due date works correctly

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-TFS-005

Test Case Title: Verify that users can save and apply custom filter and sort settings

Test Case Description: This test case verifies that users can save custom filter and sort settings and apply them later to quickly view tasks based on their preferences.

Test Suite: Task Filtering and Sorting

Test Priority: High

Preconditions:

- User is logged in

Test Data: No test data needed

Test Steps:

1. Navigate to the task list page
2. Apply various filters (e.g., due date, priority, assignee) and sorts (e.g., due date, priority, assignee) to the task list
3. Click the "Save Filter Settings" button
4. Provide a name for the custom filter settings
5. Save the filter
6. Apply the saved filter settings by selecting the name from the list of saved filters

Postconditions:

- Task list updates to display tasks matching the saved filter settings

Expected Result:

- The task list displays tasks matching the saved filter settings, indicating that users can save and apply custom filter and sort settings correctly

Severity: Major

Priority: High

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: TC001\_TimeTracker\_StartStopTimer

Test Case Title: Verify that users can start and stop a timer for each task

Test Case Description: This test case verifies that users can successfully start and stop a timer for individual tasks to accurately track time spent on those tasks.

Test Suite: Time Tracking Functionality

Test Priority: High

Preconditions:

- User must be logged in
- User must be on the task details page



Test Data: No test data needed

Test Steps:

1. Log in as a user
2. Navigate to a task details page
3. Click the "Start Timer" button
4. Wait for a few minutes
5. Click the "Stop Timer" button

Postconditions:

- The system displays the total time spent on the task, including the started and stopped timer intervals

Expected Result: The test case passes if the user can start and stop the timer for a task, and the system accurately records and displays the total time spent on the task.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC002\_TimeTracker\_ManualTimeEntry

Test Case Title: Verify that users can manually enter or adjust time entries

Test Case Description: This test case verifies that users can manually enter or adjust time spent on tasks when the timer function is not suitable or preferred.

Test Suite: Time Tracking Functionality

Test Priority: Medium

Preconditions:

- User must be logged in
- User must be on the task details page

Test Data: Time data for manual entry (e.g., "2h 30m")

Test Steps:

1. Log in as a user
2. Navigate to a task details page
3. Locate the manual time entry field
4. Enter the time data (e.g., "2h 30m")
5. Save the changes

Postconditions:

- The system stores the entered time data and displays it on the task details page

Expected Result: The test case passes if the user can manually enter time data for a task, and the system accurately stores and displays the entered time on the task details page.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC003\_TimeTracker\_TimeDataExport

Test Case Title: Verify that time tracking data is stored and can be exported

Test Case Description: This test case verifies that time tracking data is properly stored and can be exported in a suitable format for further analysis or reporting.

Test Suite: Time Tracking Data Handling

Test Priority: Medium

Preconditions:

- User must be logged in
- User must have tracked time for at least one task

Test Data: No test data needed

Test Steps:

1. Log in as a user
2. Navigate to the time tracking data page
3. Initiate the export process for time tracking data (e.g., CSV, Excel)
4. Save the exported file

Postconditions:

- The system generates the time tracking data file in the selected format
- The generated file contains accurate time tracking data for all tasks

Expected Result: The test case passes if the user can successfully export time tracking data in a suitable format, and the exported file contains accurate time tracking data for all tasks.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC004\_TimeTracker\_TimeReports

Test Case Title: Verify that the system provides reports on time spent per task or project

Test Case Description: This test case verifies that the system offers comprehensive reports on time spent for individual tasks or projects, enabling efficient resource allocation and project management.

Test Suite: Time Tracking Reporting

Test Priority: Medium

Preconditions:

- User must be logged in
- User must have tracked time for at least one task

Test Data: No test data needed

Test Steps:

1. Log in as a user
2. Navigate to the time tracking reports page
3. Generate a report for a specific task or project

Postconditions:

- The system generates a time report for the specified task or project
- The report contains accurate time data for the selected task or project

Expected Result: The test case passes if the user can generate a time report for a specific task or project, and the report contains accurate time data for the selected item.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC005\_TimeTracker\_TimerEdgeCase

Test Case Title: Verify that the timer handles edge cases such as overlapping intervals and timer resumption

Test Case Description: This test case verifies that the timer handles edge cases such as overlapping intervals and timer resumption to ensure accurate time tracking and prevent potential data inconsistencies.

Test Suite: Time Tracking Functionality (Edge Cases)

Test Priority: Low

Preconditions:

- User must be logged in
- User must be on the task details page

Test Data: Multiple timer intervals for a single task

#### Test Steps:

1. Log in as a user
2. Navigate to a task details page
3. Start the timer
4. Stop the timer after a few minutes
5. Repeat steps 3-4 multiple times, creating overlapping intervals
6. Resume the timer from a previously stopped interval

#### Postconditions:

- The system accurately displays the total time spent on the task, accounting for overlapping intervals and resumed timers

Expected Result: The test case passes if the timer handles edge cases such as overlapping intervals and timer resumption, and the system accurately displays the total time spent on the task.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Negative

-----

#### **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: Verify the functionality that allows users to create a new task template, input necessary information, and save it for future use.

Test Suite: Task Template Creation

Test Priority: High

Preconditions:

- User is logged into the Task Management Tool
- User has appropriate permissions to create task templates

Test Data: New task template details (name, predefined task structure, description, assignees, due dates)

Test Steps:

1. Navigate to the task templates section.
2. Click on the "Create New Template" button.

3. Enter the new task template details.
4. Save the new task template.
5. Verify that the new task template is displayed in the list of available templates.

Postconditions:

- New task template is saved and accessible in the list of available templates

Expected Result: New task template is successfully created, saved, and displayed in the list of available templates.

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: Verify the ability to configure task templates with predefined task details, descriptions, assignees, and due dates.

Test Suite: Task Template Configuration

Test Priority: High

Preconditions:

- User is logged into the Task Management Tool
- User has appropriate permissions to create task templates

Test Data: Task template details (name, predefined task structure, description, assignees, due dates)

Test Steps:

1. Navigate to the task templates section.
2. Click on the "Create New Template" button.

3. Enter a new task template name.
4. Define the predefined task structure (e.g., subtasks, checklists).
5. Add a task description.
6. Assign at least one user to the task template.
7. Set due dates for the task template.
8. Save the new task template.

Postconditions:

- New task template is saved with predefined task details, descriptions, assignees, and due dates

Expected Result: Task template is successfully created with predefined task structures, descriptions, assignees, and due dates.

Severity: Major

Priority: High

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: Verify the process of applying a saved task template to a new task or project.

Test Suite: Task Template Application

Test Priority: High

Preconditions:

- User is logged into the Task Management Tool
- User has appropriate permissions to create tasks or projects
- At least one task template exists

Test Data: Selected task template, new task or project

Test Steps:



1. Navigate to the task or project creation section.
2. Select a task template from the available templates list.
3. Create a new task or project.
4. Verify that the new task or project contains the predefined task structure, descriptions, assignees, and due dates from the selected template.

Postconditions:

- New task or project is created with predefined task details from the selected template

Expected Result: Selected task template is successfully applied to a new task or project, with predefined task structures, descriptions, assignees, and due dates.

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: Verify the ability to modify or remove saved task templates.

Test Suite: Task Template Modification

Test Priority: Medium

Preconditions:

- User is logged into the Task Management Tool
- User has appropriate permissions to edit or delete task templates
- At least one task template exists

Test Data: Selected task template

Test Steps:

1. Navigate to the task templates section.

2. Select a task template to edit.
3. Modify the template details (name, predefined task structure, description, assignees, due dates).
4. Save the updated template.
5. Verify that the changes are reflected in the updated task template.
6. Select the same task template to delete.
7. Delete the selected task template.
8. Verify that the task template is no longer displayed in the list of available templates.

Postconditions:

- Task template is successfully updated with new details
- Task template is successfully deleted from the list of available templates

Expected Result: Selected task template can be successfully edited and deleted, reflecting the desired updates or removal from the list of available 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: Verify the functionality that displays the list of available task templates for application to new tasks or projects.

Test Suite: Task Template Display

Test Priority: Medium

Preconditions:

- User is logged into the Task Management Tool
- At least one task template exists

Test Steps:

1. Navigate to the task or project creation section.
2. Locate the list of available task templates.
3. Verify that at least one task template is displayed in the list.

Expected Result: A list of available task templates is displayed for selection during task or project creation, containing at least one previously saved template.

Severity: Major

Priority: Medium

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: TCT-001

Test Case Title: Verify that users can sync tasks with external calendars (e.g., Google Calendar, Outlook)

Test Case Description: As a quality assurance engineer, I need to verify that users can successfully connect their Task Management Tool account with external calendar services and synchronize their tasks.

Test Suite: Calendar Integration

Test Priority: High

Preconditions:

- User has a valid account on the Task Management Tool.
- User is logged in to the Task Management Tool.
- User has a valid account with external calendar service (e.g., Google Calendar, Outlook).
- User has granted necessary permissions for the Task Management Tool to access their external calendar.

Test Data: User accounts, external calendar accounts, tasks with deadlines

Test Steps:

1. Log in to the Task Management Tool as a user.
2. Access the calendar integration settings.
3. Choose the external calendar service to connect with.
4. Follow the prompts to grant permissions for the Task Management Tool to access the user's external calendar.
5. Create a task with a deadline and save it.
6. Verify that the task appears in the user's external calendar.

Postconditions:

- Task appears in the user's external calendar.
- The connection between the Task Management Tool and the user's external calendar remains active.

Expected Result: The task created on the Task Management Tool appears in the user's external calendar.

Severity: Critical

Priority: High

Type of Testing: Functional Testing, Integration Testing

Test Case Approach: Positive

Test Case ID: TCT-002

Test Case Title: Verify that the system displays task deadlines in the calendar view

Test Case Description: As a quality assurance engineer, I need to verify that task deadlines are displayed correctly in the calendar view of the Task Management Tool.

Test Suite: Calendar Integration

Test Priority: Medium

Preconditions:

- User has a valid account on the Task Management Tool.
- User is logged in to the Task Management Tool.
- User has at least one task with a deadline.

Test Data: User accounts, tasks with deadlines

Test Steps:

1. Log in to the Task Management Tool as a user.
2. Navigate to the calendar view.
3. Verify the presence of task deadlines within the calendar view.
4. Check the accuracy of task deadline dates and times in the calendar view.

Postconditions:

- Task deadlines are correctly displayed in the calendar view.

Expected Result: Task deadlines are accurately represented in the calendar view.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, System Testing

Test Case Approach: Positive

Test Case ID: TCT-003

Test Case Title: Verify that users can set reminders for tasks via the calendar

Test Case Description: As a quality assurance engineer, I need to confirm that users can create reminders for tasks using the calendar integration within the Task Management Tool.

Test Suite: Calendar Integration

Test Priority: Medium

Preconditions:

- User has a valid account on the Task Management Tool.
- User is logged in to the Task Management Tool.
- User has at least one task with a deadline.
- User has connected their Task Management Tool account to an external calendar service (e.g., Google Calendar, Outlook).

Test Data: User accounts, tasks with deadlines, external calendar accounts

Test Steps:

1. Log in to the Task Management Tool as a user.
2. Create a task with a deadline.
3. Access the task's settings.
4. Set a reminder for the task using the calendar integration.
5. Verify the presence of the reminder for the task in the user's external calendar.

Postconditions:

- A reminder is set for the task.

- The reminder appears in the user's external calendar.

Expected Result: The reminder for the task is accurately displayed in the user's external calendar.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Integration Testing

Test Case Approach: Positive

Test Case ID: TCT-004

Test Case Title: Verify that calendar events update automatically when tasks are changed

Test Case Description: As a quality assurance engineer, I need to confirm that changes made to tasks in the Task Management Tool are automatically reflected in the user's external calendar.

Test Suite: Calendar Integration

Test Priority: High

Preconditions:

- User has a valid account on the Task Management Tool.
- User is logged in to the Task Management Tool.
- User has a valid account with an external calendar service (e.g., Google Calendar, Outlook).
- User has connected their Task Management Tool account to an external calendar service.
- User has at least one task with a deadline.

Test Data: User accounts, tasks with deadlines, external calendar accounts

Test Steps:

1. Log in to the Task Management Tool as a user.
2. Access the task's settings.
3. Change the deadline for the task.
4. Verify that the deadline change is propagated to the user's external calendar.
5. Verify that any associated reminders are updated in the user's external calendar.

Postconditions:

- The deadline change is correctly represented in the user's external calendar.
- Any associated reminders are updated accordingly.

Expected Result: Changes made to tasks are automatically updated in the user's external calendar.

Severity: Critical

Priority: High

Type of Testing: Functional Testing, Integration Testing

Test Case

Approach: Positive

Test Case ID: TCT-005

Test Case Title: Verify that users can manage calendar sync settings within the app

Test Case Description: As a quality assurance engineer, I need to confirm that users have access to and can manage calendar sync settings within the Task Management Tool.

Test Suite: Calendar Integration

Test Priority: Low

Preconditions:

- User has a valid account on the Task Management Tool.
- User is logged in to the Task Management Tool.

Test Data: User accounts

Test Steps:

1. Log in to the Task Management Tool as a user.
2. Access the calendar integration settings.
3. Verify the presence of sync settings for external calendar services.
4. Update synced calendar services or disable/enable calendar sync.
5. Verify that the changes to the sync settings are saved and applied.

Postconditions:



- Calendar sync settings have been updated.
- The Task Management Tool reflects these changes.

Expected Result: Calendar integration settings can be managed by users within the Task Management Tool.

Severity: Major

Priority: Low

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: TC-TMT-TD-001

Test Case Title: Verify that users can duplicate tasks.

Test Case Description: This test case aims to verify that the system allows users to create a duplicate of an existing task.

Test Suite: Task Duplication

Test Priority: High

Preconditions:

- User is logged in.
- A task exists and is visible to the user.

Test Data: Existing task with details, descriptions, assignees, and deadlines.

Test Steps:

1. Navigate to the task list.
2. Select an existing task.
3. Click on the "Duplicate" button.
4. Provide a name for the duplicated task.
5. Confirm the duplication action.

Postconditions:

- A new task is created.
- The new task has the same details, descriptions, assignees, and deadlines as the original task.

Expected Result: A duplicate task is created with all the details from the original task.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-TD-002

Test Case Title: Verify that the duplicated task retains all original details.

Test Case Description: This test case aims to ensure that the duplicated task preserves all the details from the original task, including descriptions, assignees, and deadlines.

Test Suite: Task Duplication

Test Priority: High

Preconditions:

- User is logged in.
- A task exists and is visible to the user.
- A duplicate task has been created using TC-TMT-TD-001.

Test Data: Duplicate task with details, descriptions, assignees, and deadlines.

Test Steps:

1. Navigate to the task list.
2. Select the duplicated task.
3. Verify that the task details, descriptions, assignees, and deadlines match those of the original task.

Postconditions:

- Duplicated task details match the original task.

Expected Result: The duplicated task has all the same details, descriptions, assignees, and deadlines as the original task.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-TD-003

Test Case Title: Verify that users can modify the duplicated task.

Test Case Description: This test case aims to confirm that users can edit or modify the details of a duplicated task after it has been created.

Test Suite: Task Duplication

Test Priority: Medium

Preconditions:

- User is logged in.
- A duplicate task has been created using TC-TMT-TD-001.

Test Data: Duplicate task with details, descriptions, assignees, and deadlines.

Test Steps:

1. Navigate to the task list.
2. Select the duplicated task.
3. Edit the task details (e.g., change the name, assignee, or deadline).
4. Save the changes.

Postconditions:

- Duplicated task details have been updated.

Expected Result: The duplicated task details can be modified by the user.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-TD-004

Test Case Title: Verify that the system indicates a task has been duplicated.

Test Case Description: This test case aims to confirm that the system displays a clear and understandable message when a task is duplicated, informing the user of the successful duplication action.

Test Suite: Task Duplication

Test Priority: Low

Preconditions:

- User is logged in.
- A task exists and is visible to the user.

Test Data: Existing task with details, descriptions, assignees, and deadlines.

Test Steps:

1. Navigate to the task list.
2. Select an existing task.
3. Click on the "Duplicate" button.
4. Verify the message indicating the duplication action was successful.

Postconditions:

- Duplicated task is saved and displayed in the task list.
- Message informing the user of the successful duplication action is displayed.

Expected Result: The system shows a clear and understandable message indicating that the task has been duplicated successfully.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-TD-005

Test Case Title: Verify that duplicated tasks are saved and displayed in the task list.

Test Case Description: This test case aims to ensure that the duplicated task is saved and displayed in the task list accordingly.

Test Suite: Task Duplication

Test Priority: High

Preconditions:

- User is logged in.
- A task exists and is visible to the user.
- A duplicate task has been created using TC-TMT-TD-001.

Test Data: Duplicate task with details, descriptions, assignees, and deadlines.

Test Steps:

1. Navigate to the task list.
2. Verify the duplicated task displays in the list.

Postconditions:

- Duplicated task is visible in the task list.

Expected Result: The duplicated task is displayed in the task list with all the details and configurations from the original task.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

-----

**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 1:

Test Case ID: TCTMTH-001

Test Case Title: Verify that users can view a detailed history of changes to tasks

Test Case Description: As a user, I want to verify that I can access and view the detailed task history, including all modifications, comments, and other relevant activities.

Test Suite: Task Management Tool - Task History

Test Priority: High

Preconditions:

- User must be logged in to the system
- A task with multiple updates and modifications must exist

Test Data: No test data needed

Test Steps:

1. Navigate to the task details page
2. Locate and click on the "Task History" tab
3. Verify that the history view displays all changes made to the task, including modifications, comments, and

other relevant activities

Postconditions:

- The task history is displayed correctly
- The system maintains a complete audit trail of task-related actions

Expected Result:

- The test case passes if the task history shows all necessary information, ensuring accountability and transparency.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TCTMTH-002

Test Case Title: Verify that users can filter the history by type of change or date

Test Case Description: As a user, I want to verify that I can filter the task history to display modifications, comments, or other activities by type or date.

Test Suite: Task Management Tool - Task History

Test Priority: Medium

Preconditions:

- User must be logged in to the system
- A task with multiple updates and modifications must exist

Test Data: No test data needed

Test Steps:

1. Navigate to the task details page
2. Locate and click on the "Task History" tab
3. Apply a filter for a specific type of change (e.g., modification or comment) or date range



4. Verify that the task history displays only the selected filter criteria

Postconditions:

- The filtering functionality works as expected
- Users can efficiently track specific types of changes or activity within a given date range

Expected Result:

- The test case passes if users can successfully apply filters and the task history displays the correct data based on the filter criteria.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TCTMTH-003

Test Case Title: Verify that the system provides timestamps for all changes

Test Case Description: As a user, I want to verify that the task history contains accurate and consistent timestamps for all task-related actions.

Test Suite: Task Management Tool - Task History

Test Priority: Medium

Preconditions:

- User must be logged in to the system
- A task with multiple updates and modifications must exist

Test Data: No test data needed

Test Steps:

1. Navigate to the task details page
2. Locate and click on the "Task History" tab
3. Scroll through the task history, verifying that each action has an accurate and consistent timestamp

Postconditions:

- The system maintains accurate time-tracking for all changes
- Users can efficiently monitor the progression of task development

Expected Result:

- The test case passes if each task-related action in the history contains accurate and consistent timestamps.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TCTMTH-004

Test Case Title: Verify that task history is accessible from the task details page

Test Case Description: As a user, I want to verify that task history is easily accessible from the task details page to ensure quick and efficient access to the audit trail of modifications and other activities related to the task.

Test Suite: Task Management Tool - Task History

Test Priority: Low

Preconditions:

- User must be logged in to the system
- At least one task must be created

Test Data: No test data needed

Test Steps:

1. Navigate to the task details page
2. Locate and click on the "Task History" tab

Postconditions:

- The system provides a direct link to the task history from the task details page

- Users can efficiently access the task history

Expected Result:

- The test case passes if the task history is accessible from the task details page within one click.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TCTMTH-005

Test Case Title: Verify that task history remains consistent and accurate when multiple users access and modify the task

Test Case Description: As a user, I want to verify that the task history maintains consistent and accurate records when multiple users access and modify the task, ensuring transparency and accountability across all team members.

Test Suite: Task Management Tool - Task History

Test Priority: High

Preconditions:

- Multiple users must be logged in to the system
- A task with multiple updates and modifications must exist

Test Data: No test data needed

Test Steps:

1. Assign the task to multiple users
2. Instruct the users to make modifications and comments
3. Navigate to the task details page and open the task history
4. Verify that the history records all updates and modifications, including the users responsible for each change and the accurate timestamps

Postconditions:

- The system accurately records the changes made by multiple users
- Task history maintains a consistent and accurate audit trail

Expected Result:

- The test case passes if the task history accurately reflects the updates and modifications made by multiple users.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Negative

-----

**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: As a user, I want to receive email notifications when there is an update on a task I am assigned to, so that I can stay informed about ongoing task progress.

Test Suite: Task Notifications

Test Priority: High

Preconditions:

- User is registered and logged in
- User has at least one task assigned to them
- User has enabled email notifications in their settings

Test Data: Task with an update (e.g. status change, priority change, new comment)

Test Steps:

1. Log in as a registered user
2. Access the task list or dashboard
3. Identify a task with an update
4. Check for a new email notification in the user's inbox

Postconditions:

- User has received an email notification related to the updated task

Expected Result:

- Email notification contains detailed information about the task update (what has changed, who made the change, etc.)

Severity: Major

Priority: High

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

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that users can customize notification settings

Test Case Description: As a user, I want to be able to customize my notification settings based on my preferences, so that I can choose how I want to receive notifications.

Test Suite: Task Notifications

Test Priority: Medium

Preconditions:

- User is registered and logged in

Test Data: User account with notification settings to configure

Test Steps:

1. Log in as a registered user
2. Navigate to the notification settings page
3. Select preferred notification delivery methods (email, mobile push, in-app) and frequencies
4. Save the updated settings

Postconditions:

- User's notification settings have been saved and updated

Expected Result:

- User receives notifications according to their customized settings

Severity: Major

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: As a user, I want notifications to include relevant and detailed information about the activity, so that I can quickly understand the context and take appropriate action if needed.

Test Suite: Task Notifications

Test Priority: High

Preconditions:

- User is registered and logged in
- User has at least one task assigned to them

Test Data: Task with an update (e.g. status change, priority change, new comment)

Test Steps:

1. Log in as a registered user
2. Access the task list or dashboard
3. Identify a task with an update
4. Check for a new in-app notification or mobile push notification
5. Read the notification details

Postconditions:

- User has received a notification with detailed information about the task update

Expected Result:

- Notification contains clear and concise details about the task update, including what has changed, who made the change, and any additional context

Severity: Major

Priority: High

Type of Testing: Functional Testing, Integration Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that the system logs notification history

Test Case Description: As a user, I want the system to maintain a log of all notifications for future reference, so that I can review past notifications and track task activity.

Test Suite: Task Notifications

Test Priority: Medium

Preconditions:

- User is registered and logged in

Test Data: User account with notification history

Test Steps:

1. Log in as a registered user
2. Access the notification history page
3. Review the list of past notifications

Postconditions:

- User can view a comprehensive log of past notifications

Expected Result:

- Notification log contains all notifications related to the user's tasks, including date, time, and details about the activity that triggered the notification

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, System Testing

Test Case Approach: Positive

Test Case ID: TC005



Test Case Title: Verify that users do not receive duplicate notifications

Test Case Description: As a user, I want to avoid receiving duplicate notifications, so that my inbox or device remains manageable and clutter-free.

Test Suite: Task Notifications

Test Priority: Low

Preconditions:

- User is registered and logged in
- User has at least one task assigned to them

Test Data: Task with an update (e.g. status change, priority change, new comment)

Test Steps:

1. Log in as a registered user
2. Access the task list or dashboard
3. Identify a task with an update
4. Check for a new in-app notification or mobile push notification
5. Wait for an email notification

Postconditions:

- User has received only one notification about the task update

Expected Result:

- The system does not send duplicate notifications via different channels for the same task update

Severity: Minor

Priority: Low

Type of Testing: Functional Testing, Non-Functional Testing, Regression Testing

Test Case Approach: Negative, Destructive

-----

**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: TCT-TASKMGT-ARCH-001

Test Case Title: Verify that users can archive completed tasks

Test Case Description: As a user, I want to archive a completed task so that I can keep my task list focused on active tasks.

Test Suite: Task Archiving

Test Priority: High

Preconditions:

1. User is logged in.
2. A task has been created and marked as complete.

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

Test Steps:

1. Navigate to the task list.
2. Locate the completed task.
3. Click on the "Archive" button next to the task.
4. Confirm the archiving action in the pop-up window.

Postconditions:

1. The task is moved to the archived section.
2. The task list is updated to show only active tasks.

Expected Result: The specified completed task is successfully archived and moved to a separate section for easy access.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-TASKMGT-ARCH-002

Test Case Title: Verify that archived tasks are moved to a separate section

Test Case Description: As a user, I want archived tasks to be stored in a separate section so that I can easily access them when needed.

Test Suite: Task Archiving

Test Priority: High

Preconditions:

1. User is logged in.
2. A task has been archived.

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

Test Steps:

1. Navigate to the task list.
2. Switch to the archived section view.
3. Locate the archived task.

Postconditions:

1. The archived task is displayed in the archived section.

Expected Result: The archived task is successfully moved to a separate section, and users can easily access and view it.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-TASKMGT-ARCH-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 so that I can access historical data.

Test Suite: Task Archiving

Test Priority: Medium

Preconditions:

1. User is logged in.
2. Tasks have been archived.

Test Data: Multiple archived tasks with varying titles, descriptions, due dates, and categories.

Test Steps:

1. Navigate to the task list.
2. Switch to the archived section view.

3. Use the search bar to search for an archived task using a keyword from its title or description.

Postconditions:

1. The searched archived task is displayed.

Expected Result: Users can search for and view archived tasks using keywords from their titles or descriptions.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-TASKMGT-ARCH-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 to be able to restore archived tasks to the active list so that I can reactivate them if needed.

Test Suite: Task Archiving

Test Priority: Medium

Preconditions:

1. User is logged in.

2. A task has been archived.

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

Test Steps:

1. Navigate to the task list.

2. Switch to the archived section view.

3. Locate the archived task.

4. Click on the "Restore" button next to the task.

5. Confirm the restoring action in the pop-up window.

Postconditions:

1. The task is moved to the active task list.

Expected Result: The specified archived task is successfully restored to the active task list.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-TASKMGT-ARCH-005

Test Case Title: Verify that archived tasks retain all original details and history

Test Case Description: As a user, I want archived tasks to maintain their original details and history so that I can access the full context when needed.

Test Suite: Task Archiving

Test Priority: Medium

Preconditions:

1. User is logged in.
2. A task has been archived.

Test Data: Archived task with a title, description, due date, attachments, comments, and category.

Test Steps:

1. Navigate to the task list.
2. Switch to the archived section view.
3. Locate the archived task.
4. View the details (title, description, due date, attachments, comments, and category) of the archived task.

Postconditions:

1. All the original details and history of the archived task are displayed.

Expected Result: Archived tasks retain all original details and history, allowing users to access the full context.

Severity: Major

Priority: Medium

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: TC-GA-001

Test Case Title: Verify that users can invite external collaborators as guests with limited access.

Test Case Description: This test case verifies the user's ability to send an invitation to an external collaborator, granting them guest access to specific tasks or projects within the Task Management Tool.

Test Suite: Guest Access

Test Priority: High

Preconditions:

- User is logged in
- User has a task or project to which they want to grant guest access

Test Data: External collaborator's email address

Test Steps:

1. Navigate to the task or project you wish to share
2. Select the "Guest Access" option
3. Input the external collaborator's email address
4. Click "Send Invitation"

Postconditions:

- Invitation email has been sent to the external collaborator

Expected Result: Invitation email successfully sent; external collaborator can follow the link to register as a guest user and access the specified tasks or projects.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive



Test Case 2:

Test Case ID: TC-GA-002

Test Case Title: Verify that guests can view and collaborate on specific tasks or projects.

Test Case Description: This test case checks if a guest user can view and interact with the tasks or projects they have been granted access to by a full-access user.

Test Suite: Guest Access

Test Priority: Medium

Preconditions:

- Guest user has accepted the invitation
- Guest user is logged in

Test Data: No test data needed

Test Steps:

1. Log in as a guest user
2. Navigate to the shared task or project
3. Verify that all task/project information is available and up-to-date
4. Attempt to edit a task within the shared project
5. Verify that the edit functionality is available for collaboration
6. Save the edited task

Postconditions:

- Guest user has viewed and edited the shared task

Expected Result: Guest user can view and collaborate on the shared tasks or projects with the expected level of access.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-GA-003

Test Case Title: Verify that the system provides clear distinctions between guest and full access roles.

Test Case Description: This test case confirms that the Task Management Tool clearly distinguishes between user roles and that guests can only access specified tasks or projects.

Test Suite: Guest Access

Test Priority: Medium

Preconditions:

- Full-access user is logged in
- Guest user has accepted the invitation and is logged in

Test Data: No test data needed

Test Steps:

1. Log in as a full-access user
2. Create a task not shared with the guest user
3. Log in as the guest user
4. Attempt to view the non-shared task
5. Check if the system restricts access to the non-shared task

Postconditions:

- Guest user cannot view the non-shared task

Expected Result: The system clearly differentiates between guest and full-access user roles and prevents guests from accessing sensitive or restricted areas.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Negative

Test Case 4:

Test Case ID: TC-GA-004

Test Case Title: Verify that users can manage and revoke guest access at any time.

Test Case Description: This test case ensures that full-access users have the ability to manage and revoke guest access to tasks and projects when necessary.

Test Suite: Guest Access

Test Priority: High

Preconditions:

- Full-access user is logged in
- Guest user has accepted the invitation and is logged in

Test Data: No test data needed

Test Steps:

1. Log in as a full-access user
2. Navigate to the task or project shared with the guest
3. Revoke the guest user's access to the task or project
4. Log in as the guest user
5. Attempt to access the revoked task or project

Postconditions:

- Guest user cannot access the revoked task or project

Expected Result: Full-access users can manage and revoke guest access as needed, preventing unauthorized access.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Negative

Test Case 5:

Test Case ID: TC-GA-005

Test Case Title: Verify that after revoking access, the guest cannot access sensitive or restricted areas of the platform.

Test Case Description: This test case confirms the revoking of a guest user's access prevents them from accessing any restricted areas of the platform.

Test Suite: Guest Access

Test Priority: High

Preconditions:

- Guest user has been revoked from a task or project
- Guest user is logged in

Test Data: No test data needed

Test Steps:

1. Log in as the revoked guest user
2. Attempt to access any non-shared tasks or projects
3. Attempt to access administrative or sensitive features

Postconditions:

- Guest user cannot access non-shared tasks or projects
- Guest user cannot access administrative or sensitive features

Expected Result: After revoking a guest user's access, they are unable to access any sensitive or restricted areas of the platform.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Negative

-----

**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 1:

Test Case ID: TC\_TMT\_IE\_001

Test Case Title: Verify that users can import tasks from a CSV file

Test Case Description: This test case verifies the functionality of importing tasks from a CSV file, ensuring that the system correctly maps fields and allows users to preview imported data before finalizing the process.

Test Suite: Task Import

Test Priority: High

Preconditions:

- User is logged in
- A valid CSV file containing task data is prepared

Test Data: Prepared CSV file with task data

Test Steps:

1. Navigate to the task import page
2. Select CSV as the file format
3. Click on "Choose File" and select the prepared CSV file
4. Click on "Import"
5. Review the mapped fields and modify if necessary
6. Preview the imported tasks
7. Click on "Confirm Import"

Postconditions:

- Imported tasks should be visible in the task list
- System logs should contain an entry for the import activity

Expected Result:

- The system should import tasks from the CSV file successfully
- Fields should be mapped correctly, and users should be able to modify mappings if necessary
- Users should be able to preview imported tasks before finalizing the import

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC\_TMT\_IE\_002

Test Case Title: Verify that users can export tasks to an Excel file

Test Case Description: This test case verifies the functionality of exporting tasks to an Excel file, ensuring that the system logs the export activity.

Test Suite: Task Export

Test Priority: High

Preconditions:

- User is logged in
- At least one task exists in the task list

Test Data: No test data needed

Test Steps:

1. Navigate to the task list page
2. Select at least one task from the task list
3. Click on "Export Tasks"
4. Choose Excel as the file format
5. Click on "Export"

Postconditions:

- An Excel file containing the selected tasks should be downloaded
- System logs should contain an entry for the export activity

Expected Result:

- The system should export tasks to an Excel file successfully
- Downloaded Excel file should contain the selected tasks

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC\_TMT\_IE\_003

Test Case Title: Verify that the system logs import/export activities

Test Case Description: This test case verifies that the system logs import and export activities for future reference.

Test Suite: Task Import/Export Logs

Test Priority: Medium

Preconditions:

- User is logged in
- At least one import or export activity is performed

Test Data: No test data needed

Test Steps:

1. Perform an import or export activity

Postconditions:

- System logs should contain an entry for the import or export activity

Expected Result:

- The system should log import and export activities for future reference

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC\_TMT\_IE\_004

Test Case Title: Verify that users cannot preview imported data if an invalid file format is used

Test Case Description: This test case verifies that users cannot preview imported data if an invalid file format is used for importing tasks.



Test Suite: Task Import

Test Priority: Low

Preconditions:

- User is logged in
- An invalid file format containing task data is prepared

Test Data: Prepared invalid file format with task data

Test Steps:

1. Navigate to the task import page
2. Select an invalid file format as the file format
3. Click on "Choose File" and select the prepared invalid file
4. Click on "Import"

Postconditions:

- Imported tasks should not be visible in the task list
- System logs should contain an entry for the import activity (with a warning or error message)

Expected Result:

- The system should not import tasks from the invalid file format
- Users should not be able to preview imported tasks from the invalid file format

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Negative

Test Case 5:

Test Case ID: TC\_TMT\_IE\_005

Test Case Title: Verify that users can import tasks from an Excel file

Test Case Description: This test case verifies the functionality of importing tasks from an Excel file, ensuring that the system correctly maps fields and allows users to preview imported data before finalizing the process.

Test Suite: Task Import

Test Priority: High

Preconditions:

- User is logged in
- A valid Excel file containing task data is prepared

Test Data: Prepared Excel file with task data

Test Steps:

1. Navigate to the task import page
2. Select Excel as the file format
3. Click on "Choose File" and select the prepared Excel file
4. Click on "Import"
5. Review the mapped fields and modify if necessary
6. Preview the imported tasks
7. Click on "Confirm Import"

Postconditions:

- Imported tasks should be visible in the task list
- System logs should contain an entry for the import activity

Expected Result:

- The system should import tasks from the Excel file successfully
- Fields should be mapped correctly, and users should be able to modify mappings if necessary
- Users should be able to preview imported tasks before finalizing the import

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

-----

**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: TC-TMT-001

Test Case Title: Verify that users can set task priorities

Test Case Description: Verify that registered users can set task priorities (low, medium, high) during task creation and editing.

Test Suite: Task Management Suite

Test Priority: High

Preconditions:

- User is registered and logged in
- User is on the task creation or editing page

Test Data: New or existing task with appropriate priority levels (low, medium, high)

Test Steps:

1. Navigate to the task creation or editing page.
2. Enter a new task or edit an existing task.
3. Select a priority level (low, medium, high) from the priority dropdown menu.
4. Save or update the task.

Postconditions:

- The new or updated task is saved with the selected priority.

Expected Result: The system allows users to set task priorities during task creation or editing, saving the tasks with the selected priorities.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-002

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

Test Case Description: Verify that users can sort and filter tasks based on priority levels in the task management view.

Test Suite: Task Management Suite

Test Priority: High

Preconditions:

- User is registered and logged in

- User has created tasks with different priority levels

Test Data: A list of tasks with varying priority levels (low, medium, high)

Test Steps:

1. Log in and navigate to the task management view.
2. Ensure that the sorting and filtering options are available.
3. Select the "Sort by Priority" option.
4. Verify that tasks are sorted by priority (low to high or high to low).
5. Use the filtering option to view tasks based on priority levels.

Postconditions:

- Tasks are sorted and filtered based on priority levels.

Expected Result: The system permits users to sort and filter tasks based on priority levels in the task management view.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-003

Test Case Title: Verify that the system provides visual indicators of task priority

Test Case Description: Verify that the task management view displays visual indicators for task priority levels.

Test Suite: Task Management Suite

Test Priority: Medium

Preconditions:

- User is registered and logged in
- User has created tasks with different priority levels

Test Data: A list of tasks with varying priority levels (low, medium, high)

Test Steps:

1. Log in and navigate to the task management view.
2. Verify that tasks are labeled or have icons indicating their priority levels.

Postconditions:

- Visual indicators are present for task priority levels.

Expected Result: The system displays visual indicators (labels or icons) for task priority levels in the task management view.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-004

Test Case Title: Verify that users receive notifications for high-priority tasks

Test Case Description: Verify that users receive notifications (email or in-app) for high-priority tasks nearing their deadlines.

Test Suite: Notification Suite

Test Priority: Medium

Preconditions:

- User is registered and logged in
- User has created tasks with high priority levels and deadlines

Test Data: High-priority tasks with deadlines within the notification timeframe

Test Steps:

1. Log in and ensure that notifications are enabled.
2. Create high-priority tasks with deadlines within the notification timeframe.
3. Wait for the notification timeframe to pass.

Postconditions:

- User receives notifications for high-priority tasks.

Expected Result: The system sends notifications (email or in-app) for high-priority tasks nearing their deadlines.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TMT-005

Test Case Title: Verify that priority changes are logged and tracked

Test Case Description: Verify that changes to task priorities are recorded in the system, including the timestamp and user who made the change.

Test Suite: Audit Log Suite

Test Priority: Low

Preconditions:

- User is registered and logged in
- User has the permission to edit task priorities

Test Data: A task with an existing priority level

Test Steps:

1. Log in and navigate to the task management view.
2. Edit a task's priority.
3. Verify that the priority change is recorded in the audit log with the user and timestamp.

Postconditions:

- Priority changes are logged in the audit log.

Expected Result: The system tracks and records priority changes, including the user and timestamp, in the audit log.

Severity: Major

Priority: Low

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: TCT-TMT-001

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

Test Case Description: As a user, I want to assign a task to a team member to facilitate collaboration and enable them to contribute to the task.

Test Suite: Task Management Suite

Test Priority: High

Preconditions:

- User is logged in
- A project or category exists
- User has create and assign task permissions

Test Data: A new task with a description, deadline, and relevant attachments; at least one team member associated with the project or category

Test Steps:

1. Navigate to the project or category
2. Click on the "Create Task" button
3. Enter task details, including description, deadline, and attachments
4. Select the team member from the list of available members
5. Save the new task

Postconditions:

- The new task is created and assigned to the selected team member
- The team member receives a notification of the task assignment

Expected Result: The assigned team member can view, edit, and update the task, contributing to the project's success.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Integration Testing

Test Case Approach: Positive

Test Case ID: TCT-TMT-002

Test Case Title: Verify that users can share updates and communicate through comments or messages

Test Case Description: As a user, I want to share updates and communicate about tasks with my team members to keep everyone informed and engaged.

Test Suite: Collaboration Suite

Test Priority: Medium

Preconditions:

- User is logged in
- A project or category exists
- User has create and edit task permissions
- A task exists with at least one assigned team member

Test Data: A task update, comment, or message to share with team members

Test Steps:

1. Navigate to the specific task
2. Click on the "Add Comment" or "Send Message" button
3. Enter the update or message and select the intended recipients
4. Send the update or message

Postconditions:

- The update or message is visible to the assigned team members
- Team members receive notifications about the update or message

Expected Result: Team members can engage in productive discussions, share insights, and work together effectively.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Integration Testing

Test Case Approach: Positive

Test Case ID: TCT-TMT-003

Test Case Title: Verify that task updates and comments are visible to assigned team members

Test Case Description: As a user, I want to ensure that task updates and comments made by team members are visible to me, so that I can stay informed and respond accordingly.

Test Suite: Collaboration Suite

Test Priority: Medium

Preconditions:

- User is logged in
- A project or category exists
- User has create and edit task permissions
- A task exists with at least one assigned team member
- Another team member has made updates or comments on the task

Test Data: None

Test Steps:

1. Navigate to the specific task
2. Review task details, updates, and comments

Postconditions:

- All updates, comments, and task details are visible to the user

Expected Result: The user can view all task-related information, respond to comments, and collaborate effectively with their team.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Integration Testing

Test Case Approach: Positive

Test Case ID: TCT-TMT-004

Test Case Title: Verify that users receive notifications for task updates and comments

Test Case Description: As a user, I want to be notified of task updates and comments from team members so that I can stay informed and participate in the collaboration process.

Test Suite: Notification Suite

Test Priority: Medium

Preconditions:

- User is logged in
- A project or category exists
- User has create and edit task permissions
- A task exists with at least one assigned team member
- Another team member has made updates or comments on the task

Test Data: None

Test Steps:

1. Log in as the user
2. Wait for notifications related to the task updates and comments

Postconditions:

- User receives notifications for task updates and comments

Expected Result: Users are promptly informed about task-related activities, enabling them to engage in timely collaboration.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Integration Testing

Test Case Approach: Positive

Test Case ID: TCT-TMT-005

Test Case Title: Verify that the system provides a clear view of task collaboration history

Test Case Description: As a user, I want to have access to a comprehensive history of task collaboration,

enabling me to track progress and review past discussions.

Test Suite: Collaboration Suite

Test Priority: Low

Preconditions:

- User is logged in
- A project or category exists
- User has create and edit task permissions
- A task exists with a history of updates, comments, and assignments

Test Data: None

Test Steps:

1. Navigate to the specific task
2. Access the collaboration history (e.g., comments, updates, assigned team members)

Postconditions:

- A complete history of task collaboration is visible and accessible

Expected Result: Users can review task collaboration history, track progress, and gain insights from past discussions.

Severity: Major

Priority: Low

Type of Testing: Functional Testing, Integration 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: Validate the task tracking feature by viewing and confirming the accuracy of task statuses for various tasks in different projects.

Test Suite: Task Management - Task Tracking

Test Priority: High

Preconditions:

- User must be logged in
- User must have access to at least one project
- At least one task must be created within the project

Test Data: Tasks with varying statuses within the project

Test Steps:

1. Navigate to the project dashboard
2. Select a specific project to view tasks
3. Locate and identify tasks within the project
4. Verify each task's status by checking the status indicator
5. Manually update a task's status and verify the change

Postconditions:

- Project tasks are displayed correctly
- Task statuses are updated appropriately

Expected Result: All task statuses are accurately reflected and can be viewed by 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 tasks are updated in real-time

Test Case Description: Validate the task tracking feature by verifying that tasks are updated in real-time for all users.

Test Suite: Task Management - Task Tracking

Test Priority: High

Preconditions:

- User must be logged in
- User must be part of a shared project
- Another user must have access to the shared project
- At least one task must be created within the project

Test Data: Tasks with varying statuses within the project

Test Steps:

1. Navigate to the project dashboard
2. Create or update a task within the project
3. Notify another user about the task update
4. Instruct the other user to refresh their browser or check the task in real-time
5. Verify if the task update is displayed for the other user

Postconditions:

- Task updates are visible for all users in real-time

Expected Result: Task updates are displayed immediately for all users within the project.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-003

Test Case Title: Verify that system provides visual indicators for task progress

Test Case Description: Validate the task tracking feature by confirming that the system provides visual indicators for task progress, such as percent complete or progress bars.

Test Suite: Task Management - Task Tracking

Test Priority: High

Preconditions:

- User must be logged in
- User must have access to at least one project
- At least one task must be created within the project

Test Data: Tasks with varying progress percentages within the project

Test Steps:

1. Navigate to the project dashboard
2. Select a specific project to view tasks



3. Locate and identify tasks within the project
4. Verify the visual indicator for each task's progress, such as a progress bar or percent complete
5. Manually update a task's progress and verify the visual indicator change

Postconditions:

- Visual indicators accurately portray the progress of tasks

Expected Result: The system provides accurate visual indicators for the progress of tasks.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-004

Test Case Title: Verify that notifications and reminders are sent for upcoming deadlines

Test Case Description: Validate the task tracking feature by confirming that notifications and reminders are sent for upcoming task deadlines.

Test Suite: Task Management - Task Tracking

Test Priority: Medium

Preconditions:

- User must be logged in
- User must have access to at least one project
- At least one task with a deadline must be created within the project

Test Data: Tasks with varying deadlines within the project

Test Steps:

1. Navigate to the project dashboard
2. Select a specific project to view tasks
3. Locate and identify tasks within the project
4. Verify that no reminders have been sent for tasks with far deadlines

5. Approach the deadline for a task

6. Verify that a notification or reminder has been sent for the approaching deadline

Postconditions:

- Reminders are sent at appropriate times before task deadlines

Expected Result: The system sends reminders for approaching task deadlines at appropriate times.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-005

Test Case Title: Verify that tasks cannot be marked as complete if the deadline has not passed

Test Case Description: Validate the task tracking feature by confirming that tasks cannot be marked as complete if the deadline has not passed.

Test Suite: Task Management - Task Tracking

Test Priority: Medium

Preconditions:

- User must be logged in
- User must have access to at least one project
- At least one task with a deadline must be created within the project

Test Data: Tasks with deadlines within the project

Test Steps:

1. Navigate to the project dashboard
2. Select a specific project to view tasks
3. Locate and identify tasks within the project
4. Attempt to mark a task as complete before its deadline
5. Check for error messages or failed submission

Postconditions:

- Task remains in its current status

Expected Result: Users cannot mark tasks as complete before their deadlines.

Severity: Major

Priority: Medium

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: TC-TN-001

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

Test Case Description: This test case verifies that the system sends a notification to the user when a new task is created and assigned to them.

Test Suite: Task Notifications

Test Priority: High

Preconditions:

- User is registered and logged into the system.
- A project or category exists.

Test Data:

- Task details, including project/category, title, description, assignee, and deadline.

Test Steps:

1. Log into the system as an assignee.
2. Access the project or category where the new task will be created.
3. Create a new task with the provided test data.
4. Ensure the task is saved and assigned correctly.

Postconditions:

- A notification is displayed for the user in the system.
- The user receives an email notification (if enabled).

Expected Result: The user receives a notification for the new task in the system and via email (if enabled).

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TN-002

Test Case Title: Verify that users receive notifications for task updates and comments

Test Case Description: This test case checks that the system sends a notification to the user when a task assigned to them is updated or when a comment is added.

Test Suite: Task Notifications

Test Priority: High

Preconditions:

- User is registered and logged into the system.
- A task exists and is assigned to the user.

Test Data:

- Updated task details, including title, description, assignee, and deadline (if updated).
- Comment content for the task.

Test Steps:

1. Log into the system as the task assignee.
2. Access the task to be updated or commented on.
3. Update the task or add a comment using the provided test data.
4. Save the changes.

Postconditions:

- A notification is displayed for the user in the system.
- The user receives an email notification (if enabled).

Expected Result: The user receives a notification for the task update or comment in the system and via email (if enabled).

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TN-003

Test Case Title: Verify that users receive reminders for upcoming deadlines

Test Case Description: This test case ensures that the system sends a reminder notification to the user when a task deadline is near.

Test Suite: Task Notifications

Test Priority: High

Preconditions:

- User is registered and logged into the system.
- A task exists, is assigned to the user, and has a deadline within the reminder period.

Test Data:

- Task details, including project/category, title, description, assignee, and deadline (within the reminder period).

Test Steps:

1. Log into the system as the task assignee.
2. Access the task with the upcoming deadline.

Postconditions:

- A notification is displayed for the user in the system before the deadline.
- The user receives an email notification (if enabled).

Expected Result: The user receives a reminder notification in the system and via email (if enabled) before the task deadline.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TN-004

Test Case Title: Verify that notification settings can be customized by users

Test Case Description: This test case verifies that users can customize their notification preferences for tasks and projects.

Test Suite: Task Notifications

Test Priority: Medium

Preconditions:

- User is registered and logged into the system.

Test Data:

- Notification settings, such as email notifications, desktop notifications, mobile push notifications, and frequency (instant or daily digest).

Test Steps:

1. Log into the system as a user.
2. Access the user profile settings.
3. Update the notification preferences using the provided test data.
4. Save the changes.

Postconditions:

- Notification settings are updated for the user.

Expected Result: The user can customize their notification preferences for tasks and projects, and the changes are saved successfully.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-TN-005

Test Case Title: Verify that users do not receive notifications when notifications are disabled

Test Case Description: This test case confirms that the system does not send notifications when a user has

disabled them.

Test Suite: Task Notifications

Test Priority: Low

Preconditions:

- User is registered and logged into the system.
- A task exists and is assigned to the user.

Test Data:

- Notification settings, such as email notifications, desktop notifications, and mobile push notifications, set to disabled.

Test Steps:

1. Log into the system as a user.
2. Access the user profile settings.
3. Disable all notification preferences using the provided test data.
4. Save the changes.
5. Create a new task or update an existing task.
6. Add a comment to the task.

Postconditions:

- Notifications are disabled for the user.

Expected Result: The user does not receive any notifications for tasks, updates, or comments when notifications are disabled.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Negative

-----



**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: TC001

Test Case Title: Verify that users can create and assign tasks to different categories or projects

Test Case Description: This test case verifies that users can create a new task and assign it to a specific category or project during the task creation process.

Test Suite: Task Creation and Assignment

Test Priority: High

Preconditions:

- User is logged in
- User has access to create tasks
- Categories or projects have been created

Test Data: New task details, a category or project to assign the task to

Test Steps:

1. Navigate to the task creation page
2. Enter the new task details
3. Select a category or project from the dropdown list
4. Save and confirm the new task creation

Postconditions:

- New task is created and saved
- Task is assigned to the selected category or project

Expected Result: The task is successfully created and displayed under the assigned category or project in the task list.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that users can view tasks grouped by categories or projects

Test Case Description: This test case verifies that users can filter and view tasks based on their assigned categories or projects.

Test Suite: Task View and Filtering

Test Priority: Medium

Preconditions:

- User is logged in

- Tasks have been created
- Categories or projects have been assigned to tasks

Test Data: None

Test Steps:

1. Navigate to the task list page
2. Click on the filter button
3. Select a category or project from the dropdown list
4. Confirm the filter selection

Postconditions:

- Task list is updated with tasks from the selected category or project

Expected Result: The task list displays only the tasks associated with the selected category or project.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that tasks can be moved between categories or projects

Test Case Description: This test case verifies that users can move an existing task from one category or project to another.

Test Suite: Task Editing and Moving

Test Priority: Medium

Preconditions:

- User is logged in
- Tasks have been created
- Categories or projects have been assigned to tasks

Test Data: A task to be moved to a different category or project

#### Test Steps:

1. Navigate to the task list page
2. Locate the task to be moved
3. Click on the "Move" button associated with the task
4. Select a new category or project from the dropdown list
5. Confirm the move by saving changes

#### Postconditions:

- The task is moved to the new category or project

Expected Result: The task is removed from the previous category or project and added to the new one.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that category or project information is accurately reflected in task views

Test Case Description: This test case verifies that category or project information is correctly displayed and updated in various task views, such as the task list, task details, or notifications.

Test Suite: Task View and Information Accuracy

Test Priority: Low

#### Preconditions:

- User is logged in
- Tasks have been created
- Categories or projects have been assigned to tasks

Test Data: None

#### Test Steps:

1. Navigate to the task list page

2. Verify that each task displays its assigned category or project
3. Click on a task to view task details
4. Verify that category or project information is displayed correctly
5. Receive a notification for an upcoming deadline or reminder
6. Verify that category or project information is included in the notification

Postconditions:

- Category or project information is accurately displayed in task views

Expected Result: Category or project information is consistently displayed and updated in various task views.

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that tasks cannot be assigned to non-existent categories or projects

Test Case Description: This test case verifies that users cannot assign tasks to non-existent categories or projects.

Test Suite: Task Creation and Negative Scenarios

Test Priority: Low

Preconditions:

- User is logged in
- User has access to create tasks

Test Data: A non-existent category or project name

Test Steps:

1. Navigate to the task creation page
2. Enter the new task details
3. Attempt to assign the task to a non-existent category or project

#### 4. Save the new task

Postconditions:

- The new task is not saved

Expected Result: The system prevents the user from saving the task and displays an error message about the invalid category or project name.

Severity: Major

Priority: Low

Type of Testing: Negative 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: TST-001

Test Case Title: Verify that users can search for tasks by name

Test Case Description: This test case verifies that the task management tool allows users to search for tasks based on their names, ensuring that the search functionality returns accurate and relevant results.

Test Suite: Task Search

Test Priority: High

Preconditions:

- User is logged in
- Task(s) with unique names exist in the system

Test Data: A set of tasks with distinct names

Test Steps:

1. Log in as a user
2. Navigate to the task search page
3. Enter the name of a task in the search bar
4. Click the 'Search' button
5. Review the list of search results

Postconditions:

- User remains logged in

Test Data: N/A

Expected Result: The search results should display the task(s) with the specified name.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TST-002

Test Case Title: Verify that users can search for tasks by description

Test Case Description: This test case validates that the task management tool permits users to search for tasks using a keyword in their descriptions, ensuring that the search returns accurate and relevant results.

Test Suite: Task Search

Test Priority: High

Preconditions:

- User is logged in
- Task(s) with unique descriptions exist in the system

Test Data: A set of tasks with distinct descriptions

Test Steps:

1. Log in as a user
2. Navigate to the task search page
3. Enter a keyword from a task description in the search bar
4. Click the 'Search' button
5. Review the list of search results

Postconditions:

- User remains logged in

Test Data: N/A

Expected Result: The search results should display the task(s) that contain the keyword in their descriptions.

Severity: Major

Priority: High

Type of Testing: Functional Testing



Test Case Approach: Positive

Test Case ID: TST-003

Test Case Title: Verify that users can filter search results by category, priority, or due date

Test Case Description: This test case checks if the task management tool allows users to filter search results based on categories, priorities, or due dates, ensuring that the filtering functionality works as expected.

Test Suite: Task Search

Test Priority: High

Preconditions:

- User is logged in
- Task(s) with various categories, priorities, and due dates exist in the system

Test Data: A set of tasks with different categories, priorities, and due dates

Test Steps:

1. Log in as a user
2. Navigate to the task search page
3. Enter a task name or keyword in the search bar
4. Click the 'Search' button
5. Apply filters using category, priority, or due date options
6. Review the list of filtered search results

Postconditions:

- User remains logged in

Test Data: N/A

Expected Result: The search results should be refined according to the selected filters, displaying only the tasks that match the specified criteria.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TST-004

Test Case Title: Verify that search results are accurate and relevant

Test Case Description: This test case aims to validate that the task management tool returns accurate and relevant search results based on user input, ensuring the system's consistency and integrity.

Test Suite: Task Search

Test Priority: High

Preconditions:

- User is logged in
- A sufficient number of tasks with different attributes exist in the system

Test Data: A set of tasks with various attributes

Test Steps:

1. Log in as a user
2. Navigate to the task search page
3. Enter different task names, descriptions, categories, priorities, and due dates as search queries
4. Click the 'Search' button after each query
5. Review the list of search results

Postconditions:

- User remains logged in

Test Data: N/A

Expected Result: The search results should accurately reflect the user's input, displaying only the tasks that match the specified criteria.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TST-005

Test Case Title: Verify that users can handle search input errors gracefully

Test Case Description: This test case verifies that the task management tool handles user input errors in the search bar gracefully, ensuring a good user experience and minimal inconvenience.

Test Suite: Task Search

Test Priority: Medium

Preconditions:

- User is logged in

Test Data: Invalid search inputs, such as symbols, numbers, or special characters

Test Steps:

1. Log in as a user
2. Navigate to the task search page
3. Enter invalid search inputs in the search bar
4. Click the 'Search' button
5. Observe and record any error messages or behavior

Postconditions:

- User remains logged in

Test Data: N/A

Expected Result: The system should display an appropriate error message informing users about the invalid input and prompting them to enter a valid search query.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

## 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 ID: TCT-TMT-001

Test Case Title: Verify that users can add comments to tasks

Test Case Description: This test case aims to confirm that the system allows users to add comments to tasks,

facilitating collaboration and communication among team members.

Test Suite: Task Collaboration

Test Priority: High

Preconditions:

- User must be logged in
- A task must exist

Test Data: No test data needed

Test Steps:

1. Navigate to the task details page
2. Locate the comment section
3. Enter a comment in the text field
4. Click the 'Add Comment' button
5. Verify the new comment appears in the comment section

Postconditions:

- The new comment is saved and visible to all users with access to the task

Expected Result: The test case is considered a success if the user can add a comment to a task and it is visible to other users.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-TMT-002

Test Case Title: Verify that users can attach files to tasks

Test Case Description: This test case verifies that users are able to attach files to tasks, allowing for better collaboration and information sharing.

Test Suite: Task Collaboration

Test Priority: High

Preconditions:

- User must be logged in
- A task must exist

Test Data: Test file for attachment

Test Steps:

1. Navigate to the task details page
2. Locate the file attachment section
3. Click on the 'Attach File' button
4. Select a test file from local storage
5. Click 'Open' to upload the file
6. Verify the file is displayed in the file attachment section

Postconditions:

- The file is saved and accessible by all users with access to the task

Expected Result: The test case is considered a success if the user can attach a file to a task and it is visible and accessible to other users.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-TMT-003

Test Case Title: Verify that users can share tasks with other users

Test Case Description: This test case validates the functionality that enables users to share tasks with other users, promoting collaboration and ensuring that the right people are involved in the task.

Test Suite: Task Collaboration

Test Priority: High

Preconditions:

- User must be logged in
- A task must exist

Test Data: User email addresses for sharing

Test Steps:

1. Navigate to the task details page
2. Locate the 'Share Task' button
3. Enter the email addresses of the users to share the task with
4. Click 'Send Invitations'
5. Verify the invited users are displayed in the 'Shared With' section

Postconditions:

- The invited users receive notifications and have access to the task

Expected Result: The test case is considered a success if the user can share a task with other users and the invited users can access the shared task.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TCT-TMT-004

Test Case Title: Verify that task collaboration features do not disrupt the system or user experience

Test Case Description: This test case checks that the task collaboration features work seamlessly and do not affect the overall system performance or user experience negatively.

Test Suite: Task Collaboration

Test Priority: Medium

Preconditions:

- User must be logged in

- At least one task with comments and attachments must exist

Test Data: No test data needed

Test Steps:

1. Navigate to the task details page
2. Add a new comment and attach a file
3. Collaborate with other users on the task
4. Monitor system resources and performance
5. Repeat steps 2-4 for multiple tasks and users

Postconditions:

- The system remains stable, and the user experience is not impacted negatively

Expected Result: The test case is considered a success if the system remains stable, and the user experience remains positive even when multiple users collaborate on tasks simultaneously.

Severity: Major

Priority: Medium

Type of Testing: Non-Functional Testing (System Performance)

Test Case Approach: Positive

Test Case ID: TCT-TMT-005

Test Case Title: Verify that user access to tasks and collaboration features is managed correctly

Test Case Description: This test case validates the proper management of user access to tasks and collaboration features based on user roles and permissions to ensure the security and integrity of the system.

Test Suite: Task Collaboration

Test Priority: Medium

Preconditions:

- Different users with various roles and permissions must exist
- At least one task with comments and attachments must exist

Test Data: No test data needed



Test Steps:

1. Log in as a user with minimal permissions
2. Attempt to access a task with comments and attachments
3. Verify that access is denied or limited according to permissions
4. Log in as a user with full permissions
5. Verify that full access to comments and attachments is granted
6. Repeat steps 2-5 for multiple users and tasks

Postconditions:

- User access is managed correctly based on roles and permissions

Expected Result: The test case is considered a success if user access to tasks and collaboration features is managed correctly based on their roles and permissions.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing (Security)

Test Case Approach: Positive & Negative

-----

**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: TC-CC-001

Test Case Title: Verify that instructors can create courses using videos

Test Case Description: The test case aims to verify that an instructor can create a course by uploading a video as a course element.

Test Suite: Course Creation Tools

Test Priority: High

Preconditions:

- User is logged in as an instructor
- User has access to the course creation dashboard

Test Data:

- A sample video file (MP4 format, less than 1 GB)

Test Steps:

1. Log in as an instructor
2. Navigate to the course creation dashboard

3. Click on "Add new course"
4. Enter course details
5. In the "Course Content" section, click on "Add Content"
6. Select "Video"
7. Click "Upload Video" and select the sample video file
8. Assign a title and description for the video
9. Save changes

Postconditions:

- The course should be saved in the drafts
- The video should be successfully uploaded and displayed in the course content

Expected Result:

- Course creation should be successful with the video content

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-CC-002

Test Case Title: Verify that course preview is displayed correctly

Test Case Description: The test case aims to validate that the course preview displays the content properly before publishing.

Test Suite: Course Preview and Publishing

Test Priority: Medium

Preconditions:

- User is logged in as an instructor
- User has created a course with multiple content types (video, presentation, quiz, and assignment)

Test Data: N/A

Test Steps:

1. Log in as an instructor
2. Navigate to the course creation dashboard
3. Open the draft course
4. Click on "Preview Course"

Test Steps for each content type:

1. Play the video
2. Navigate through the slides of the presentation
3. Attempt a quiz question and verify the correctness and error messages
4. View and download the assignment

Postconditions:

- Course preview should display all content types

Expected Result:

- The course preview displays all content types in a sequential and interactive manner

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-CC-003

Test Case Title: Verify that instructors can save courses as drafts

Test Case Description: The test case aims to confirm that instructors can save their courses as drafts before publishing.

Test Suite: Course Creation Tools

Test Priority: Low

Preconditions:

- User is logged in as an instructor
- User has access to the course creation dashboard

Test Data:

- A sample course with some content

Test Steps:

1. Log in as an instructor
2. Navigate to the course creation dashboard
3. Create a new course and add content
4. Click on "Save As Draft"

Postconditions:

- The course should be saved in the instructor's drafts list

Expected Result:

- The course is saved correctly as a draft, allowing further editing

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-CC-004

Test Case Title: Verify that courses can be edited post-publishing

Test Case Description: The test case ensures that instructors can edit the published courses.

Test Suite: Course Editing

Test Priority: Medium

Preconditions:

- User is logged in as an instructor

- User has published a course

Test Data:

- Changes to be made in the published course

Test Steps:

1. Log in as an instructor
2. Navigate to the course creation dashboard
3. Open the published course
4. Edit the content or course details
5. Save changes

Postconditions:

- Course content and details should be updated

Expected Result:

- The published course should be editable by the instructor and updated accordingly

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC-CC-005

Test Case Title: Verify that the tools are user-friendly and require minimal technical skills

Test Case Description: The test case evaluates the user-friendliness of the course creation tools and the technical skill required for using them.

Test Suite: Usability and Technical Skills

Test Priority: High

Preconditions:

- User has an average proficiency in basic computer operations

Test Data:

- A novice instructor with limited technical knowledge

Test Steps:

1. Provide a novice instructor with the course creation dashboard
2. Instruct the instructor to create a course using the available tools
3. Observe the time taken for completion and any challenges faced
4. Document difficulties and required assistance

Expected Result:

- The instructor should be able to create a course within a reasonable time, with minimal difficulties or assistance and intuitively navigate the interface

Severity: Blocker

Priority: High

Type of Testing: Usability Testing

Test Case Approach: Negative (focus on highlighting difficulties)

-----

**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: TC-IL-001

Test Case Title: Verify that lessons include quizzes, polls, simulations, and discussions

Test Case Description: Validate the presence of different interactive elements like quizzes, polls, simulations, and discussions within the lesson content.

Test Suite: Interactive Lessons

Test Priority: High

Preconditions:

- User is logged into the learning platform
- User has access to a course with interactive lessons

Test Data: No test data needed

Test Steps:

1. Navigate to a course with interactive lessons
2. Access each lesson within the course
3. Check for the presence of quizzes, polls, simulations, and discussions

Postconditions:

- Interactive elements are confirmed to be present and functional



Expected Result: All four interactive elements (quizzes, polls, simulations, and discussions) are available and functional within the lesson content.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-IL-002

Test Case Title: Verify that interactive elements are integrated seamlessly into the lesson content

Test Case Description: Assess the integration of interactive components into lessons to ensure they are cohesive and do not disrupt the learning experience.

Test Suite: Interactive Lessons

Test Priority: Medium

Preconditions:

- User is logged into the learning platform
- User has access to a course with interactive lessons

Test Data: No test data needed

Test Steps:

1. Navigate to a course with interactive lessons
2. Access each lesson containing interactive elements
3. Evaluate the cohesiveness and smoothness of interactive components within the lesson

Postconditions:

- Interactive elements are confirmed to be integrated seamlessly and do not negatively impact the learning experience

Expected Result: Interactive components are smoothly and cohesively integrated into the lesson content without causing disruptions or confusion.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Usability Testing

Test Case Approach: Positive

Test Case ID: TC-IL-003

Test Case Title: Verify that learners can track their progress through these activities

Test Case Description: Evaluate the progress tracking feature for interactive lessons to ensure it accurately reflects learners' engagement with quizzes, polls, simulations, and discussions.

Test Suite: Interactive Lessons

Test Priority: High

Preconditions:

- User is logged into the learning platform
- User has access to a course with interactive lessons

Test Data: Learner engagement with multiple interactive elements

Test Steps:

1. Navigate to a course with interactive lessons
2. Engage with various interactive elements as a learner
3. Review the progress tracking feature to assess the learner's progress

Postconditions:

- Progress tracking feature accurately reflects the learner's engagement with interactive elements

Expected Result: Progress tracking feature accurately reflects the learner's progress with interactive elements, accounting for completed quizzes, polls, simulations, and discussion contributions.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: TC-IL-004

Test Case Title: Verify that feedback is provided after each interactive component

Test Case Description: Confirm that learners receive feedback upon completion of quizzes, polls, simulations, and discussions.

Test Suite: Interactive Lessons

Test Priority: High

Preconditions:

- User is logged into the learning platform
- User has access to a course with interactive lessons

Test Data: Learner engagement with multiple interactive elements

Test Steps:

1. Navigate to a course with interactive lessons
2. Engage with various interactive components as a learner
3. Complete each interactive component

Postconditions:

- Learner receives feedback for each interactive element

Expected Result: Learner receives feedback for each interactive element, such as quiz scores, poll results, simulation outcomes, and discussion contributions.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: TC-IL-005

Test Case Title: Verify that interactive lessons can be accessed on various devices

Test Case Description: Test the accessibility of interactive lessons on different devices (desktop, tablet, mobile) to ensure consistent user experience.

Test Suite: Interactive Lessons

Test Priority: High

Preconditions:

- User has access to the learning platform

Test Data: No test data needed

Test Steps:

1. Access the learning platform on various devices (desktop, tablet, mobile)
2. Navigate to a course with interactive lessons
3. Engage with various interactive elements to evaluate functionality

Postconditions:

- Interactive lessons are fully functional and accessible on all tested devices

Expected Result: Interactive lessons are fully functional and accessible on all tested devices, providing a consistent user experience across platforms.

Severity: Major

Priority: High

Type of Testing: Compatibility Testing, Functional 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 Case Description: As a learner, I want to be able to select courses from a catalog that aligns with my interests so that I can focus on topics I enjoy and am passionate about.

Test Suite: Course Selection

Test Priority: High

Preconditions:

- Learner is registered and logged into the platform
- A variety of courses and modules are available in the course catalog

Test Data: Learner with defined interests

Test Steps:

1. Login as a learner
2. Navigate to the course catalog
3. Filter courses by learner interests
4. Select a course that matches the learner's interests
5. View course details

Postconditions:

- Selected course is added to the learner's personal learning path
- No other courses are affected

Expected Result:

- Learner can successfully filter courses by interests
- Selected course is relevant to the learner's interests

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 Case Description: As a learner, I want my learning path to automatically update when I complete or fail modules, so that it reflects my current skill level and progress.

Test Suite: Progress Tracking & Adaptation

Test Priority: High

Preconditions:

- Learner is registered, logged in, and has a personal learning path
- Learner has started a course with multiple modules

Test Data: Learner progress data

Test Steps:

1. Login as a learner
2. Access the personal learning path
3. Begin a course and complete several modules
4. Fail one or more modules in the course
5. Return to the personal learning path

Postconditions:

- Completed modules are marked as finished
- Failed modules are marked as such and may require re-taking
- Remaining modules are re-ordered based on skill level and progress

Expected Result:

- Learning path updates to reflect learner progress
- Learner can see the impact of completion and failure on their learning path

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 Case Description: As a learner, I want to receive recommendations for courses and modules that align with my interests and learning objectives, so that I can continue expanding my knowledge in areas that interest me.

Test Suite: Course Recommendations

Test Priority: High

Preconditions:

- Learner is registered, logged in, and has a personal learning path
- Learner has completed at least one course

Test Data: Learner profile and completion data

Test Steps:

1. Login as a learner
2. Access the personal learning path
3. Complete a course
4. View recommended courses and modules

Postconditions:

- Recommended courses and modules align with the learner's interests
- Recommendations consider the learner's completed courses

Expected Result:

- Learner sees relevant recommendations for new courses and modules

Severity: Major

Priority: High

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 Case Description: As a learner, I want to track my progress through my personal learning path, so that I can stay motivated and monitor my educational growth.

Test Suite: Progress Tracking & Reporting

Test Priority: High

Preconditions:

- Learner is registered, logged in, and has a personal learning path
- Learner has started a course with multiple modules

Test Data: Learner progress data

Test Steps:



1. Login as a learner
2. Access the personal learning path
3. Start a course with multiple modules
4. Complete several modules in the course
5. View progress tracking data

Postconditions:

- Progress tracking accurately reflects modules started and completed

Expected Result:

- Learner sees personalized progress tracking data
- Data includes courses and modules in the personal learning path

Severity: Major

Priority: High

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 Case Description: As a learner, I want to be able to modify my learning path and choose different courses or modules as my interests change or if I need to adjust my learning objectives.

Test Suite: Learning Path Adjustment

Test Priority: High

Preconditions:

- Learner is registered, logged in, and has a personal learning path
- Learner has started a course with multiple modules

Test Data: Learner profile and learning path data

Test Steps:

1. Login as a learner

2. Access the personal learning path
3. Remove a module from the learning path
4. Add a new module to the learning path
5. Save changes

Postconditions:

- Removed module is no longer part of the learning path
- New module is added to the learning path

Expected Result:

- Learner can successfully modify their learning path
- Changes are saved and reflected in the learning path

Severity: Major

Priority: High

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: PT-001

Test Case Title: Verify that progress tracking shows completed activities and grades

Test Case Description: This test case verifies that learners can view their completed activities and corresponding grades in the progress tracking feature.

Test Suite: Progress Tracking

Test Priority: High

Preconditions:

- User is logged in
- User is enrolled in a course

Test Data: Course with multiple activities and assigned grades

Test Steps:

1. Log in as a registered user
2. Navigate to the enrolled course
3. Access the progress tracking feature
4. Observe the list of activities under the course

5. Verify that completed activities are marked as such

6. Check the corresponding grades for each activity

Postconditions:

- The system maintains an accurate record of completed activities and grades

Expected Result:

- Completed activities are displayed with corresponding grades in the progress tracking feature.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: PT-002

Test Case Title: Verify that learners can view detailed reports on their progress

Test Case Description: This test case ensures that learners can access and view a detailed report of their progress, showing advanced insights and analytics.

Test Suite: Progress Tracking

Test Priority: High

Preconditions:

- User is logged in
- User is enrolled in a course

Test Data: Course with multiple activities and assigned grades

Test Steps:

1. Log in as a registered user
2. Navigate to the enrolled course
3. Access the progress tracking feature
4. Click on the option to view the detailed progress report
5. Check the displayed data, including total activities, completed activities, and grades

Postconditions:

- The system generates a detailed progress report for the learner

Expected Result:

- A comprehensive progress report is displayed with all required information.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: PT-003

Test Case Title: Verify that achievements are highlighted in the progress tracker

Test Case Description: This test case validates that achievements earned throughout a course are highlighted and easily identifiable in the progress tracker.

Test Suite: Progress Tracking

Test Priority: Medium

Preconditions:

- User is logged in
- User is enrolled in a course with achievements

Test Data: Course with assigned achievements

Test Steps:

1. Log in as a registered user
2. Navigate to the enrolled course
3. Access the progress tracking feature
4. Locate the highlighted achievements throughout the progress tracker

Postconditions:

- Achievements are displayed with unique visual cues

Expected Result:

- Earned achievements are prominently displayed with visual indicators in the progress tracker.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: PT-004

Test Case Title: Verify that learners receive notifications about their progress

Test Case Description: This test case checks that users receive notifications regarding their progress at regular intervals and upon reaching certain milestones.

Test Suite: Progress Tracking

Test Priority: Medium

Preconditions:

- User is logged in
- User is enrolled in a course

Test Data: Course with multiple activities and assigned grades

Test Steps:

1. Log in as a registered user
2. Navigate to the enrolled course
3. Access the progress tracking feature
4. Ensure that progress notifications are enabled
5. Perform various activities in the course
6. Observe notifications related to the performed activities

Postconditions:

- Progress notifications are accurately sent

Expected Result:

- Learners receive notifications related to their progress in the course, including completed activities and

earned achievements.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: PT-005

Test Case Title: Verify that progress tracking can be exported as reports

Test Case Description: This test case ensures that progress tracking data can be exported in a widely supported file format (e.g. CSV, PDF) for further analysis.

Test Suite: Progress Tracking

Test Priority: Low

Preconditions:

- User is logged in
- User is enrolled in a course

Test Data: Course with multiple activities and assigned grades

Test Steps:

1. Log in as a registered user
2. Navigate to the enrolled course
3. Access the progress tracking feature
4. Choose the export option to save the progress data
5. Verify that the exported data is accurate and includes all required information

Postconditions:

- Progress tracking data is successfully exported

Expected Result:

- Learners can export their progress tracking data in a supported format for offline analysis.

Severity: Trivial

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

-----

## **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: TC001\_AssessmentCreation

Test Case Title: Verify that instructors can create quizzes, tests, and assignments

Test Case Description: As an instructor, I want to create diverse assessments such as quizzes, tests, and assignments to evaluate learner comprehension and provide feedback.

Test Suite: Assessment Creation

Test Priority: High

Preconditions:

- User is logged in as an instructor
- User has access to the course they wish to create an assessment for

Test Data: New quiz/test/assignment with various question types

Test Steps:

1. Log in as an instructor
2. Navigate to the desired course
3. Select 'Add Assessment'
4. Choose quiz/test/assignment
5. Enter assessment details and add questions of various types
6. Save the assessment

Postconditions:

- Assessment is saved and viewable in the course
- Notification is displayed: "Assessment created successfully"

Expected Result: Assessment is created and saved, with the correct title and question types.

Severity: Major

Priority: High

Type of Testing: Functional Testing

## Test Case Approach: Positive

### Test Case 2:

Test Case ID: TC002\_QuestionTypes

Test Case Title: Verify that assessments can include various question types (MCQs, short answers, essays, etc.)

Test Case Description: Instructors should be able to include a variety of question types to make assessments more engaging and accurate in measuring learners' understanding.

Test Suite: Question Type Creation

Test Priority: High

#### Preconditions:

- User is logged in as an instructor
- User has access to the course they wish to add a question type to
- An assessment or a question bank exists with at least one question

Test Data: New question types (MCQs, short answers, essays, etc.)

#### Test Steps:

1. Log in as an instructor
2. Navigate to the desired course
3. Choose an existing assessment or question bank
4. Select 'Add Question'
5. Choose the desired question type (MCQ, short answer, essay, etc.)
6. Enter the question details
7. Save the question

#### Postconditions:

- Question is saved and viewable in the assessment or question bank
- Notification is displayed: "Question added successfully"

Expected Result: Question is created and saved, with the correct type and details.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC003\_AutomaticGrading

Test Case Title: Verify that automatic grading is available for objective questions

Test Case Description: Timely feedback and accurate grading are essential. Automatic grading for objective questions helps support learners in their academic development.

Test Suite: Automatic Grading

Test Priority: Medium

Preconditions:

- User is logged in as an instructor
- User has access to the course they wish to create an assessment for
- An assessment with objective questions exists

Test Data: Objective question with automatic grading enabled

Test Steps:

1. Log in as an instructor
2. Navigate to the desired course
3. Choose an existing assessment with objective questions
4. Enable automatic grading for objective questions
5. Submit assessments for grading

Postconditions:

- Objective questions are automatically graded and displayed in the grading section
- Notification is displayed: "Assessments graded successfully"

Expected Result: Objective questions are graded automatically, and the results are displayed in the grading

section.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC004\_ManualGrading

Test Case Title: Verify that manual grading is available for subjective questions

Test Case Description: Manual grading should be available for subjective questions, allowing instructors to provide valuable insights for continuous improvement.

Test Suite: Manual Grading

Test Priority: Medium

Preconditions:

- User is logged in as an instructor
- User has access to the course they wish to create an assessment for
- An assessment with subjective questions exists

Test Data: Subjective question with manual grading enabled

Test Steps:

1. Log in as an instructor
2. Navigate to the desired course
3. Choose an existing assessment with subjective questions
4. Enable manual grading for subjective questions
5. Access the grading section and enter grades and feedback for subjective questions

Postconditions:

- Grades and feedback for subjective questions are saved and displayed in the grading section
- Notification is displayed: "Assessments graded successfully"

Expected Result: Grades and feedback for subjective questions are successfully saved in the grading section.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC005\_FeedbackProvision

Test Case Title: Verify that feedback is provided to learners after grading

Test Case Description: After grading, instructors should be able to provide feedback to learners. This helps support learners in their academic development and provides valuable insights for continuous improvement.

Test Suite: Feedback Provision

Test Priority: High

Preconditions:

- User is logged in as an instructor
- User has access to the course they wish to create an assessment for
- An assessment with submitted learners' submissions exists

Test Data: Subjective questions with manual grading enabled

Test Steps:

1. Log in as an instructor
2. Navigate to the desired course
3. Choose an existing assessment with submitted learners' submissions
4. Enter grades and feedback for subjective questions
5. Save the grades and feedback

Postconditions:

- Grades and feedback for subjective questions are displayed to learners
- Notification is displayed: "Feedback provided successfully"

Expected Result: Grades and feedback for subjective questions are displayed to learners.

Severity: Major

Priority: High

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: As a learner, I want to be able to create new threads in the discussion forum by posting questions and respond to existing threads to engage in conversations with peers and instructors.

Test Suite: Discussion Forums

Test Priority: High

Preconditions:

- User is logged into the online learning platform
- User is enrolled in a course

Test Data: A question or response for an existing thread or a new thread

Test Steps:

1. Navigate to the course discussion forum
2. Select an existing thread or click "Create new thread"
3. Enter the subject and message for the post
4. Click "Post"
5. Verify that the post appears in the selected thread or as a new thread

Postconditions:

- The post is visible in the discussion forum
- The user is notified of any replies to their post

Expected Result: The user can create new threads and respond to existing threads in the discussion forum, fostering collaboration and engagement with peers and instructors.

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: As a learner or instructor, I want to be able to view and participate in threaded discussions, allowing for deeper and more structured conversations.

Test Suite: Discussion Forums

Test Priority: High

Preconditions:

- User is logged into the online learning platform
- User is enrolled in a course
- A thread exists within the discussion forum

Test Data: Replies to an existing thread

Test Steps:

1. Navigate to the course discussion forum
2. Select a thread with existing replies
3. Click "Reply" to an existing reply
4. Enter the subject and message for the reply
5. Click "Post"
6. Verify that the reply is nested correctly under the original reply

Postconditions:

- The nested reply is visible in the thread

Expected Result: The discussion forum supports threaded discussions, allowing for structured and organized conversations between learners and instructors.

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: As an instructor, I want to have the ability to moderate discussions by editing, deleting, and pinning posts, ensuring that conversations are productive and relevant.

Test Suite: Discussion Forums

Test Priority: High

Preconditions:

- User is logged into the online learning platform as an instructor
- User is associated with a course
- A thread exists within the discussion forum

Test Data: An existing post in a thread

Test Steps:

1. Navigate to the course discussion forum
2. Select a thread with an existing post
3. Click the "Edit" button next to the post
4. Edit the post and click "Save"
5. Verify that the post has been updated
6. Click the "Delete" button next to the post
7. Confirm the deletion
8. Verify that the post no longer appears in the thread
9. Pin a post by clicking the "Pin" button
10. Verify that the pinned post appears at the top of the thread

Postconditions:

- The instructor can edit, delete, and pin posts in the discussion forum

Expected Result: Instructors have the ability to moderate discussions, ensuring productive and relevant conversations within the learning community.

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: As a learner, I want to be notified of new posts and replies in the discussion forum, staying up-to-date with ongoing conversations.

Test Suite: Discussion Forums

Test Priority: Medium

Preconditions:

- User is logged into the online learning platform
- User is enrolled in a course
- Notifications are enabled for the user
- A new post or reply exists in a thread the user is participating in

Test Data: A new post or reply in a thread the user is participating in

Test Steps:

1. Navigate to the course discussion forum
2. Select a thread the user is participating in
3. Ensure that the user receives a notification for the new post or reply

Postconditions:

- The user receives a notification for the new post or reply

Expected Result: Learners receive real-time notifications for new posts and replies in the discussion forum,

promoting active engagement and participation.

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: As a user, I want to access the discussion forums on various devices, ensuring a seamless learning experience across platforms.

Test Suite: Discussion Forums

Test Priority: Low

Preconditions:

- User is logged into the online learning platform
- User is enrolled in a course

Test Data: N/A

Test Steps:

1. Access the online learning platform on a desktop computer
2. Navigate to the course discussion forum
3. Access the online learning platform on a mobile device (smartphone or tablet)
4. Navigate to the course discussion forum

Postconditions:

- The discussion forum is accessible on both desktop and mobile devices

Expected Result: The discussion forums are responsive and accessible across various devices, providing a consistent user experience.

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 ID: TC001\_LiveClassWebinar

Test Case Title: Verify that live classes and webinars are conducted through video conferencing tools

Test Case Description: As a learner, I want to ensure that live classes and webinars are conducted using video conferencing tools, so I can interact with instructors in real-time.

Test Suite: Functional Testing - Live Classes and Webinars

Test Priority: High

Preconditions:

- User must have a valid account and be logged in
- User must be enrolled in a course with live class or webinar sessions

Test Data: N/A

Test Steps:

1. Log in to the online learning platform
2. Access the course page with a scheduled live class or webinar
3. Click on the "Join Live Session" button
4. Verify that the video conferencing tool is launched (e.g., Zoom, Google Meet)
5. Confirm that audio and video are working for both instructors and learners

Postconditions: N/A

Expected Result: The live class or webinar is conducted using a video conferencing tool, with proper audio and video functionality for all participants.

Severity: Major

Priority: High

Type of Testing: Functional Testing, End-to-End Testing

Test Case Approach: Positive

Test Case ID: TC002\_RealTimeQuestions

Test Case Title: Verify that learners can ask questions in real-time during live classes and webinars

Test Case Description: As a learner, I want to be able to ask questions in real-time during live classes and

webinars, so I can clarify my doubts and engage with instructors.

Test Suite: Functional Testing - Live Classes and Webinars

Test Priority: High

Preconditions:

- User must have a valid account and be logged in
- User must be enrolled in a course with live class or webinar sessions
- User must have joined the live session using video conferencing tool

Test Data: Prepare a list of questions to ask during the live session

Test Steps:

1. Join the live class or webinar using the video conferencing tool
2. Identify an appropriate time to ask a question
3. Use the question feature within the video conferencing tool or online learning platform to pose a question
4. Verify that the question is visible to the instructor
5. Confirm that the instructor addresses the question either during the live session or afterward

Postconditions: N/A

Expected Result: Learners can ask questions in real-time during live classes and webinars using the video conferencing tool or online learning platform, and the instructor addresses the questions appropriately.

Severity: Major

Priority: High

Type of Testing: Functional Testing, End-to-End Testing

Test Case Approach: Positive

Test Case ID: TC003\_RecordedSessions

Test Case Title: Verify that sessions are recorded and available for later viewing

Test Case Description: As a learner, I want sessions to be recorded and available for later viewing, so I can review the material at my own pace.

Test Suite: Functional Testing - Live Classes and Webinars

Test Priority: High

Preconditions:

- User must have a valid account and be logged in
- User must be enrolled in a course with live class or webinar sessions

Test Data: N/A

Test Steps:

1. Join the live class or webinar using the video conferencing tool
2. Complete the live session
3. After the session, access the course page to verify if the recorded session is available
4. Play the recorded session to ensure audio, video, and content are present
5. Verify that the recorded session can be rewound, paused, and forwarded

Postconditions: N/A

Expected Result: Live class and webinar sessions are recorded and available for later viewing, providing learners with the opportunity to review the material at their own pace.

Severity: Major

Priority: High

Type of Testing: Functional Testing, End-to-End Testing

Test Case Approach: Positive

Test Case ID: TC004\_InstructorScreenSharing

Test Case Title: Verify that instructors can share their screen and resources during live classes and webinars

Test Case Description: As a learner, I want instructors to be able to share their screen and resources during live classes and webinars, so I can follow along and access relevant materials.

Test Suite: Functional Testing - Live Classes and Webinars

Test Priority: High

Preconditions:

- User must have a valid account and be logged in

- User must be enrolled in a course with live class or webinar sessions
- User must have joined the live session using the video conferencing tool

Test Data: Prepare a list of resources to share during the live session

Test Steps:

1. Join the live class or webinar using the video conferencing tool
2. Request the instructor to share their screen or resources (e.g., slides, documents)
3. Verify that the instructor's screen and resources are visible to all participants
4. Confirm that learners can view the shared resources clearly and in a timely manner
5. Verify that learners can navigate through the shared resources if necessary

Postconditions: N/A

Expected Result: During live classes and webinars, instructors can share their screen and resources, enabling learners to follow along and access relevant materials.

Severity: Major

Priority: High

Type of Testing: Functional Testing, End-to-End Testing

Test Case Approach: Positive

Test Case ID: TC005\_NotificationBeforeSession

Test Case Title: Verify that notifications are sent before sessions start

Test Case Description: As a learner, I want to receive notifications before sessions start so I can prepare and join on time.

Test Suite: Functional Testing - Live Classes and Webinars

Test Priority: Medium

Preconditions:

- User must have a valid account and be logged in
- User must be enrolled in a course with live class or webinar sessions

Test Data: N/A



Test Steps:

1. Log in to the online learning platform
2. Access the course page with a scheduled live class or webinar
3. Ensure that a notification is displayed for the upcoming live session
4. Check the notification content for accuracy (e.g., course name, session name, date, time)

Postconditions: N/A

Expected Result: Notifications are sent before live classes and webinars start, reminding learners of the upcoming sessions with accurate information.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing, End-to-End Testing, Usability 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 ID: TC1-MobileAccessibility-UI

Test Case Title: Verify that the mobile UI is optimized for touch navigation

Test Case Description: Validate the usability and responsiveness of the online learning platform's user interface on mobile devices, specifically ensuring that touch navigation elements (e.g. buttons, links, menus) are appropriately sized, spaced, and positioned for finger-based interaction.

Test Suite: Mobile User Interface

Test Priority: High

Preconditions:

1. Account creation or login is completed
2. Mobile device with touch screen is available
3. The online learning platform's mobile version is accessible

Test Data: No test data needed

Test Steps:

1. Launch the online learning platform on the mobile device
2. Navigate through the entire application using touch navigation elements
3. Zoom in and out on various pages to observe any potential issues

Postconditions:

1. No touch navigation elements have overlapping issues
2. Touch navigation elements maintain their functionality during zoom actions
3. Touch navigation elements are easily identifiable and responsive

Expected Result: The mobile UI is fully optimized for touch navigation with no defects or usability issues.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Usability Testing, Compatibility Testing

Test Case Approach: Positive

Test Case ID: TC2-MobileAccessibility-Content

Test Case Title: Verify that course materials are accessible and readable on mobile devices

Test Case Description: Assess the compatibility and readability of course materials (text, images, videos) presented within the online learning platform when accessed via smartphones and tablets.

Test Suite: Mobile Course Content

Test Priority: High

Preconditions:

1. Account creation or login is completed
2. Mobile device with internet connection is available
3. A course with materials is selected

Test Data: Selection of a diverse set of course materials (text, images, videos)

Test Steps:

1. Access a selected course on the mobile device
2. Verify the presentation of various course materials on both portrait and landscape orientations
3. Confirm that all materials load and display properly
4. Confirm that text is readable at various zoom levels
5. Validate that images are clearly visible
6. Validate that videos play without lag or display issues

Postconditions:

1. All course materials load and display as intended
2. Text and images are clear and readable
3. Videos play without experiencing display issues

Expected Result: Course materials are effectively accessible and readable on mobile devices.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Compatibility Testing

Test Case Approach: Positive

Test Case ID: TC3-MobileAccessibility-Notifications

Test Case Title: Verify that notifications are sent to mobile devices

Test Case Description: Confirm that the online learning platform sends push notifications to users' mobile devices concerning course updates (e.g. new assignments, messages, grades, etc.).

Test Suite: Mobile Notifications

Test Priority: Medium

Preconditions:

1. Account creation or login is completed
2. Mobile device with internet connection and push notification enabled is available
3. User opts into receiving notifications

Test Data: Selection of a course with updates available

Test Steps:

1. Access the account settings on the mobile device
2. Enable push notifications
3. Opt into course updates
4. Access a selected course
5. Perform an action that triggers a notification (e.g. submit a new assignment)

6. Wait for the notification to be received

Postconditions:

1. Notifications are received within a reasonable timeframe

2. Notifications accurately reflect the update

Expected Result: The online learning platform sends notifications to mobile devices for course updates.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC4-MobileAccessibility-OfflineAccess

Test Case Title: Verify that the platform supports offline access to downloaded materials

Test Case Description: Ensure the ability for users to access downloaded course materials (documents, videos, etc.) while offline using mobile devices.

Test Suite: Offline Access

Test Priority: Medium

Preconditions:

1. Account creation or login is completed

2. Mobile device with internet connection is available

3. A course with downloadable materials is selected

Test Data: Selection of a diverse set of downloadable course materials (documents, videos, etc.)

Test Steps:

1. Access a selected course on the mobile device

2. Download the desired materials for offline access

3. Disconnect the mobile device from the internet

4. Attempt to access the downloaded materials

Postconditions:

1. Downloaded materials are accessible without an internet connection
2. Materials maintain their formatting/visibility

Expected Result: The online learning platform allows offline access to downloaded materials on mobile devices.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Non-Functional Testing

Test Case Approach: Positive

Test Case ID: TC5-MobileAccessibility-NegativeUI

Test Case Title: Verify that the mobile UI handles errors appropriately

Test Case Description: Test the online learning platform's mobile UI by intentionally causing errors or misusing features, ensuring proper responses and minimal negative impact on the user experience.

Test Suite: Mobile Error Handling

Test Priority: Low

Preconditions:

1. Account creation or login is completed
2. Mobile device with internet connection is available

Test Data: N/A

Test Steps:

1. Misuse various mobile UI elements (e.g. tap rapidly in the same area, scroll too fast, etc.)
2. Attempt to access unauthorized content or actions
3. Cause actions that would typically result in errors (e.g. submit incomplete forms, attempt to access invalid links, etc.)

Postconditions:

1. No crashes have occurred
2. Error messages are clear and informative

3. The user experience remains uncompromised

Expected Result: The mobile UI handles errors and misuse appropriately with minimal impact on the user experience.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing, Non-Functional Testing

Test Case Approach: Negative

-----

**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 ID: TC-RL-001

Test Case Title: Verify that the resource library includes e-books, articles, and reference guides

Test Case Description: This test case verifies that the online learning platform's resource library contains e-books, articles, and reference guides, ensuring a comprehensive and diverse range of supplementary materials.

Test Suite: Resource Library

Test Priority: High

Preconditions:

- User is logged in
- User is on the resource library page

Test Data: N/A

Test Steps:

1. Log in as a user.
2. Navigate to the resource library page.
3. Verify that there are e-books available in the library.
4. Verify that there are articles available in the library.
5. Verify that there are reference guides available in the library.

Postconditions: N/A

Expected Result: The resource library contains e-books, articles, and reference guides.

Severity: Major

Priority: High



Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-RL-002

Test Case Title: Verify that learners can search for specific materials in the resource library

Test Case Description: This test case verifies that users can search for and locate specific resources, ensuring learners can quickly find and access the materials they need.

Test Suite: Resource Library

Test Priority: High

Preconditions:

- User is logged in
- User is on the resource library page

Test Data: Search terms (e.g., "Python programming", "Machine Learning Basics")

Test Steps:

1. Log in as a user.
2. Navigate to the resource library page.
3. Enter a search term into the search bar.
4. Click the "Search" button.
5. Verify that the search results display relevant resources based on the search term.

Postconditions: N/A

Expected Result: Users can search for and find specific resources in the library.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-RL-003

Test Case Title: Verify that resources are categorized and tagged for easy navigation

Test Case Description: This test case verifies that resources are properly categorized and tagged based on subjects and topics, ensuring learners can easily navigate and locate relevant materials.

Test Suite: Resource Library

Test Priority: Medium

Preconditions:

- User is logged in
- User is on the resource library page

Test Data: N/A

Test Steps:

1. Log in as a user.
2. Navigate to the resource library page.
3. Verify that resources are grouped into categories (e.g., programming, data science, mathematics).
4. Verify that resources have relevant tags associated with them.

Postconditions: N/A

Expected Result: Resources are categorized and tagged for easy navigation.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-RL-004

Test Case Title: Verify that learners can download resources for offline access

Test Case Description: This test case verifies that users can download resources for offline access and viewing, ensuring learners can study and review materials without internet access.

Test Suite: Resource Library

Test Priority: Medium

Preconditions:

- User is logged in
- User is on the resource library page

Test Data: Selected resource for download

Test Steps:

1. Log in as a user.
2. Navigate to the resource library page.
3. Select a resource to download.
4. Click the "Download" button.
5. Verify that the resource is downloaded and can be opened offline.

Postconditions: N/A

Expected Result: Users can download resources for offline access.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-RL-005

Test Case Title: Verify that instructors can recommend resources to learners

Test Case Description: This test case verifies that instructors can recommend resources to learners, providing personalized guidance and recommendations for supplementary materials.

Test Suite: Resource Library

Test Priority: Low

Preconditions:

- Instructor is logged in
- Instructor has access to a class or group of learners
- Resource library is populated with materials

Test Data: Selected resource for recommendation

Test Steps:

1. Log in as an instructor.
2. Navigate to the class or group of learners.
3. Select a learner.
4. Click the "Recommend Resource" button.
5. Choose a resource from the resource library.
6. Click "Confirm" to send the recommendation.

Postconditions:

- Resource recommendation is visible to the selected learner

Expected Result: Instructors can recommend resources to learners.

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: TC001

Test Case Title: Verify that learners can submit assignments for peer review

Test Case Description: This test case ensures that authenticated learners can successfully submit their course assignments for peer review.

Test Suite: Peer Review Module

Test Priority: High

Preconditions:

- Learner is logged in to the online learning platform
- The learner has an active course enrollment with assignment(s) available
- The assignment is not already submitted for peer review

Test Data: Any available course assignment not yet submitted for peer review

Test Steps:

1. Log in as a learner
2. Navigate to the active course enrollment
3. Locate the assignment that has not been submitted for peer review

4. Click on the "Submit for Peer Review" button

5. Confirm submission on the success notification or page

Postconditions:

- The submitted assignment is visible in the peer review queue
- The learner cannot submit the same assignment again

Expected Result: Assignment submission is successful, and the assignment appears in the peer review queue.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC002

Test Case Title: Verify that peers can provide feedback on submitted assignments

Test Case Description: This test case validates that authenticated learners can access the peer review queue, view assigned assignments, and provide structured feedback.

Test Suite: Peer Review Module

Test Priority: High

Preconditions:

- Learner is logged in to the online learning platform
- The learner has an active course enrollment with a submitted assignment for peer review

Test Data: Any submitted course assignment for peer review

Test Steps:

1. Log in as a learner
2. Navigate to the active course enrollment
3. Access the peer review queue from the course menu

4. Select the submitted assignment to review
5. Read the assignment instructions and view the assignment
6. Input structured feedback in the provided text area
7. Click the "Submit Feedback" button

Postconditions:

- The feedback is saved and attached to the assignment
- The reviewed assignment is no longer visible in the peer review queue

Expected Result: Feedback is successfully saved and attached to the assignment, and the assignment is no longer visible in the peer review queue.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC003

Test Case Title: Verify that reviewers remain anonymous to ensure unbiased feedback

Test Case Description: This test case guarantees that learners cannot identify the reviewer of their submitted assignments.

Test Suite: Peer Review Module

Test Priority: High

Preconditions:

- Learner is logged in to the online learning platform
- The learner has an active course enrollment with a submitted assignment for peer review
- Another authenticated learner has provided feedback on the submitted assignment

Test Data: Any submitted course assignment with peer feedback

Test Steps:

1. Log in as the learner who submitted the assignment
2. Navigate to the active course enrollment
3. Access the peer review queue from the course menu
4. Select the submitted assignment with feedback
5. Check if the feedback provides any information that identifies the reviewer

Postconditions:

- No reviewer-identifying information is present in the feedback

Expected Result: Feedback does not contain any reviewer-identifying information.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Negative (looking for the absence of specific information)

---

Test Case ID: TC004

Test Case Title: Verify that learners can view feedback on their work

Test Case Description: This test case verifies that learners can see the feedback provided on their submitted assignments in a structured format.

Test Suite: Peer Review Module

Test Priority: High

Preconditions:

- Learner is logged in to the online learning platform
- The learner has an active course enrollment with a submitted assignment and feedback

Test Data: Any submitted course assignment with peer feedback

Test Steps:



1. Log in as the learner who submitted the assignment
2. Navigate to the active course enrollment
3. Access the "My Submissions" section from the course menu
4. Select the submitted assignment with feedback
5. Check if the feedback is displayed in a structured format

Postconditions:

- Feedback is presented in a well-organized and easy-to-understand manner

Expected Result: Feedback is displayed in a structured format, allowing learners to easily review and understand the provided insights.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC005

Test Case Title: Verify that the structured review process guides the feedback

Test Case Description: This test case confirms that the peer review process includes a structured format for providing feedback.

Test Suite: Peer Review Module

Test Priority: High

Preconditions:

- Learner is logged in to the online learning platform
- The learner has an active course enrollment with a submitted assignment

Test Data: Any available course assignment not yet submitted for peer review

Test Steps:

1. Log in as a learner
2. Navigate to the active course enrollment
3. Locate the assignment that has not been submitted for peer review
4. Click on the "Submit for Peer Review" button
5. Access the feedback form after submission
6. Confirm that the feedback form includes structured fields, such as categories, ratings, or open-ended questions

Postconditions:

- The feedback form includes structured fields to guide the review process

Expected Result: The feedback form includes structured fields, such as categories, ratings, or open-ended questions, to guide the review process.

Severity: Major

Priority: Medium

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 ID: TC001

Test Case Title: Verify that learners earn certificates upon course completion.

Test Case Description: This test case verifies if a learner receives a certificate upon finishing a course on the Online Learning Platform.

Test Suite: Certificates and Badges

Test Priority: High

Preconditions:

- Learner is logged in to the platform
- Learner has enrolled in a course
- Learner has completed all lessons, quizzes, and assignments for the course

Test Data: Course: "Introduction to Programming," Learner: "John Doe"

Test Steps:

1. Log in as a learner
2. Enroll in the "Introduction to Programming" course

3. Complete all lessons, quizzes, and assignments for the course

4. Navigate to the course completion page

5. Check for a certificate download option

Postconditions:

- Certificate has been downloaded for "Introduction to Programming"

Expected Result: A certificate is available for download after completing the "Introduction to Programming" course.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that badges are awarded for specific achievements and milestones.

Test Case Description: This test case verifies if learners receive badges for specific achievements and milestones during course participation.

Test Suite: Certificates and Badges

Test Priority: High

Preconditions:

- Learner is logged in to the platform

- Learner is enrolled in a course with milestones and achievements

Test Data: Course: "Data Science Fundamentals," Learner: "Jane Doe"

Test Steps:

1. Log in as a learner

2. Enroll in the "Data Science Fundamentals" course

3. Participate in course activities to achieve milestones (e.g., completing a particular number of lessons or quizzes)

4. Verify if a badge is awarded for each milestone reached

Postconditions:

- Badges have been awarded for achieved milestones

Expected Result: A badge is awarded for each milestone reached during the "Data Science Fundamentals" course.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that certificates and badges are accessible through learner profiles.

Test Case Description: This test case tests if certificates and badges earned by learners are accessible and viewable through their profiles.

Test Suite: Certificates and Badges

Test Priority: Medium

Preconditions:

- Learner is logged in to the platform
- Learner has earned at least one certificate and one badge

Test Data: Course: "Machine Learning Essentials," Learner: "James Doe"

Test Steps:

1. Log in as a learner
2. Access the learner profile
3. Verify the presence and visibility of the certificate for "Machine Learning Essentials"
4. Verify the presence and visibility of badges for achieved milestones

Postconditions:

- Certificate and badges are accessible and visible in the learner profile

Expected Result: The certificate for "Machine Learning Essentials" and badges for achieved milestones are accessible and visible within the learner profile.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that learners can share their certificates and badges on social media.

Test Case Description: This test case verifies if learners can share their certificates and badges on various social media platforms.

Test Suite: Certificates and Badges

Test Priority: Low

Preconditions:

- Learner is logged in to the platform
- Learner has earned at least one certificate and one badge
- Learner has social media accounts (Facebook, Twitter, LinkedIn)

Test Data: Course: "Cloud Computing Basics," Learner: "Julia Doe"

Test Steps:

1. Log in as a learner
2. Access a certificate or badge from the learner profile
3. Click the "Share" button
4. Select a social media platform for sharing
5. Verify if the certificate or badge is successfully shared on the selected social media platform

Postconditions:

- Certificate or badge is shared on the selected social media platform

Expected Result: Learners can share their certificates and badges on social media platforms such as Facebook,

Twitter, or LinkedIn.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that instructors can issue custom badges for special achievements.

Test Case Description: This test case verifies if instructors can create and issue custom badges for special achievements by learners.

Test Suite: Certificates and Badges

Test Priority: Medium

Preconditions:

- Instructor is logged in to the platform
- Instructor has a course with active learners
- Instructor has the ability to create custom badges

Test Data: Course: "Web Development Advanced Techniques," Instructor: "Jake Doe," Learner: "Jessica Doe"

Test Steps:

1. Log in as an instructor
2. Access the course "Web Development Advanced Techniques"
3. Create a custom badge for a special achievement, e.g., completing a challenging project
4. Assign the custom badge to Jessica Doe for completing the challenging project
5. Verify that the custom badge is awarded and visible in Jessica Doe's profile

Postconditions:

- Custom badge is awarded and visible in Jessica Doe's profile

Expected Result: Instructors can create and issue custom badges for special achievements by learners within their courses.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

-----

**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 ID: TC001

Test Case Title: Verify that learners earn points for completing activities

Test Case Description: This test case verifies that learners are awarded points upon successful completion of activities within the online learning platform.

Test Suite: Gamification Elements

Test Priority: High

Preconditions:

- Learner must be logged in
- An activity (e.g., watching a video lecture or completing a quiz) must be available for the learner to complete

Test Data: Course with an activity assigned to it

Test Steps:

1. Log in as a learner
2. Navigate to the assigned activity within the course
3. Complete the activity according to the instructions
4. Submit the activity for review
5. Wait for the system to process the completion and assign points

Postconditions:

- Points should be displayed in the learner's profile
- The completed activity should be marked as such in the course progress

Expected Result: The learner should receive points for completing the activity, with the new point total visible in the learner's profile.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that levels are achieved based on accumulated points

Test Case Description: This test case aims to check whether learners advance through levels as they accumulate points from completing activities within the online learning platform.

Test Suite: Gamification Elements

Test Priority: Medium

Preconditions:

- Learner must be logged in
- The learner must have earned points from completing activities

Test Data: Learner with at least the minimum required points for a level promotion

Test Steps:

1. Log in as a learner
2. Navigate to the gamification dashboard
3. Verify the current level based on accumulated points
4. Perform an activity to earn additional points to promote to the next level
5. Wait for the system to process the new points

Postconditions:

- The learner's level should be updated if the new points meet the level promotion criteria

Expected Result: The learner's level should be updated based on the new points earned from completing an activity.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that leaderboards display top-performing learners

Test Case Description: This test case checks whether the leaderboards accurately rank and display the top-performing learners based on their accumulated points within the online learning platform.

Test Suite: Gamification Elements

Test Priority: Low

Preconditions:

- Multiple learners must be logged in
- Learners must have accumulated points from completing activities

Test Data: Data set consisting of multiple learners with varying point totals

Test Steps:

1. Log in as different learners or access multiple learner accounts
2. Navigate to the leaderboard section within the gamification dashboard
3. Verify the rankings and point totals of the learners to ensure that they accurately reflect the data set

Postconditions:

- The leaderboard should be updated to display the top-performing learners accurately

Expected Result: The leaderboard should accurately rank and display the top-performing learners based on their accumulated points.

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that badges are awarded for reaching milestones

Test Case Description: This test case checks whether learners receive badges for reaching specific milestones within the online learning platform.

Test Suite: Gamification Elements

Test Priority: Low

Preconditions:

- Learner must be logged in
- The learner must meet the criteria for earning a milestone badge (e.g., completing a specific number of activities)

Test Data: Learner meeting the criteria for a milestone badge

Test Steps:

1. Log in as a learner
2. Navigate to the gamification dashboard
3. Check if the milestone badge is displayed on the learner's profile

Postconditions:

- The newly earned badge should be visible on the learner's profile

Expected Result: The learner should receive the badge for reaching the specified milestone, with the badge visible on the learner's profile.

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that learners can view their progress in a gamification dashboard

Test Case Description: This test case ensures that learners can view their gamification progress (accumulated points, levels, badges, and leaderboard rank) in a dedicated gamification dashboard.

Test Suite: Gamification Elements

Test Priority: Medium

Preconditions:

- Learner must be logged in

Test Data: No test data needed

Test Steps:

1. Log in as a learner
2. Navigate to the gamification dashboard
3. Verify that the gamification dashboard displays accumulated points, levels, badges, and leaderboard rank

Postconditions:

- The gamification dashboard should present accurate information on the learner's progress

Expected Result: The learner should be able to view their progress in the gamification dashboard, including accumulated points, levels, badges, and leaderboard rank.

Severity: Major

Priority: Medium

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 ID: TC001

Test Case Title: Verify that the platform interface is available in multiple languages

Test Case Description: This test case verifies that the online learning platform interface supports multiple languages, enabling learners to use the platform in their preferred language.

Test Suite: User Interface Testing

Test Priority: High

Preconditions:

- A valid account has been created on the online learning platform.
- The user is logged into the platform.
- Multiple language options are enabled on the platform.

Test Data: User's preferred language (e.g., Spanish, French, German, etc.)

Test Steps:

1. Log into the online learning platform.
2. Access the account settings or language preference menu.
3. Select a language other than the default (e.g., English) as the preferred language.
4. Save and apply the language preference.

Postconditions:

- The platform interface is displayed in the selected language.
- All features and functions within the platform are available and operational in the chosen language.

Expected Result: The online learning platform interface is displayed in the user's preferred language without any loss of functionality or usability.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that course materials are available in different languages

Test Case Description: This test case ensures that course materials, including lecture transcripts, discussion forums, and resources, are adequately translated and available in learners' preferred languages.

Test Suite: Course Materials Testing

Test Priority: High

Preconditions:

- A valid account has been created on the online learning platform.
- The user is logged into the platform.
- A course with multilingual support is selected.

Test Data: User's preferred language (e.g., Spanish, French, German, etc.)

Test Steps:

1. Log into the online learning platform.
2. Select a course with multilingual support.
3. Access course materials, such as lecture transcripts, discussion forums, and resources.
4. Change the language preference to a language other than the default (e.g., English).

Postconditions:

- Course materials are available in the user's preferred language.

Expected Result: Course materials, including lecture transcripts, discussion forums, and resources, are adequately translated and available in the user's preferred language.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that learners can select their preferred language at any time

Test Case Description: This test case confirms that users can switch to their preferred language at any point while using the platform, ensuring uninterrupted access to educational resources.

Test Suite: User Interface Testing

Test Priority: Medium

Preconditions:

- A valid account has been created on the online learning platform.
- The user is logged into the platform.

Test Data: User's preferred language (e.g., Spanish, French, German, etc.)

Test Steps:

1. Log into the online learning platform.
2. Access any feature within the platform (e.g., courses, lectures, or quizzes).
3. Access the account settings or language preference menu.
4. Change the language preference to a language other than the default (e.g., English).
5. Save and apply the language preference.

Postconditions:

- The platform interface and all features are displayed in the selected language.

Expected Result: Users can change their preferred language seamlessly at any time while using the platform,



with the platform interface and features adapting accordingly without loss of functionality or usability.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that multilingual support includes translated help and support resources

Test Case Description: This test case validates that the platform's help and support resources, such as FAQs, guides, and support ticket options, are available in users' preferred languages.

Test Suite: Help and Support Testing

Test Priority: Medium

Preconditions:

- A valid account has been created on the online learning platform.
- The user is logged into the platform.

Test Data: User's preferred language (e.g., Spanish, French, German, etc.)

Test Steps:

1. Log into the online learning platform.
2. Access the help and support resources (e.g., FAQs, guides, or support ticket options).
3. Change the language preference to a language other than the default (e.g., English).
4. Save and apply the language preference.

Postconditions:

- Help and support resources are available in the user's preferred language.

Expected Result: The online learning platform provides multilingual help and support resources that are adequately translated and available in users' preferred languages.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that instructors can provide content in multiple languages

Test Case Description: This test case confirms that instructors can create course materials in multiple languages, enabling a diverse community of learners to access the content in their preferred language.

Test Suite: Instructor Content Creation Testing

Test Priority: Medium

Preconditions:

- A valid instructor account has been created on the online learning platform.
- The instructor is logged into the platform.

Test Data: A course with content in at least two languages (e.g., English and Spanish).

Test Steps:

1. Log into the online learning platform as an instructor.
2. Create a new course or access an existing course.
3. Create or upload course materials in at least one additional language (e.g., Spanish) besides the default (e.g., English).
4. Save and publish the course.

Postconditions:

- Course materials are available in the additional language(s).

Expected Result: Instructors can create and publish course materials in multiple languages, providing a more inclusive learning environment.

Severity: Major

Priority: Medium

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 the online learning platform can be effectively navigated and used with

popular screen reader software.

Test Suite: Accessibility Features

Test Priority: High

Preconditions:

- User account created
- Screen reader software installed

Test Data: N/A

Test Steps:

1. Log in to the online learning platform.
2. Enable the screen reader software.
3. Navigate through courses, lectures, tutorials, and quizzes using the screen reader.
4. Attempt to activate accessibility features, such as adjustable font sizes and color contrast options, using the screen reader.
5. Interact with discussion forums, progress tracking, and certification features using the screen reader.

Postconditions:

- Screen reader software has been used to navigate the platform successfully.
- All major features of the platform are accessible via the screen reader.

Expected Result: The platform is fully compatible with screen readers, allowing learners with visual impairments to access and navigate the platform effectively.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Compatibility Testing, Usability 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: Confirm that users can easily adjust the font size on the platform for improved

readability.

Test Suite: Accessibility Features

Test Priority: High

Preconditions:

- User account created

Test Data: N/A

Test Steps:

1. Log in to the online learning platform.
2. Navigate to the account settings or accessibility settings page.
3. Locate the font size adjustment option.
4. Increase and decrease the font size using the available controls.
5. Verify that content is displayed correctly with the adjusted font size.

Postconditions:

- Font size has been successfully adjusted.
- Content remains readable with the adjusted font size.

Expected Result: The platform allows users to easily adjust font sizes, improving readability for learners with visual impairments.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Usability 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: Ensure that users can adjust the color contrast on the platform for better visibility.

Test Suite: Accessibility Features

Test Priority: High

Preconditions:

- User account created

Test Data: N/A

Test Steps:

1. Log in to the online learning platform.
2. Navigate to the account settings or accessibility settings page.
3. Locate the color contrast adjustment option.
4. Increase and decrease the color contrast using the available controls.
5. Verify that content is displayed correctly with the adjusted color contrast.

Postconditions:

- Color contrast has been successfully adjusted.
- Content remains visible with the adjusted color contrast.

Expected Result: The platform allows users to easily adjust color contrast, improving visibility for learners with visual impairments.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Usability 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 easily activate and customize accessibility features on the platform.

Test Suite: Accessibility Features

Test Priority: High

Preconditions:

- User account created

Test Data: N/A

Test Steps:

1. Log in to the online learning platform.
2. Navigate to the account settings or accessibility settings page.
3. Activate font size and color contrast adjustment options.
4. Adjust font size and color contrast using the available controls.
5. Deactivate font size and color contrast adjustment options.

Postconditions:

- Accessibility features have been successfully activated and customized.
- Accessibility features can be deactivated as needed.

Expected Result: The platform provides intuitive and easy-to-use accessibility features, allowing learners with disabilities to effectively use the platform.

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: Confirm that the platform provides support resources for learners with disabilities.

Test Suite: Accessibility Features

Test Priority: Medium

Preconditions:

- User account created

Test Data: N/A

Test Steps:

1. Log in to the online learning platform.

2. Locate the support or help center link.

3. Review the available resources related to accessibility features and accommodations for learners with disabilities.

Postconditions:

- Support resources have been reviewed.

Expected Result: The platform provides comprehensive support resources for learners with disabilities, promoting equitable access to educational opportunities.

Severity: Minor

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-DIS-001

Test Case Title: Verify that instructors can edit discussion posts

Test Case Description: As an instructor, I want to edit a discussion post to correct any inaccuracies or improve the content.

Test Suite: Discussion Moderation

Test Priority: High

Preconditions:

- User is logged in as an instructor
- A discussion post exists

Test Data: A valid discussion post

Test Steps:

1. Navigate to the discussion forum
2. Locate and open the existing discussion post
3. Click on the "Edit" button below the post
4. Modify the content of the post
5. Click on the "Save" button

Postconditions:

- The discussion post has been edited with new content

Expected Result: The discussion post is successfully updated with the changes made by the instructor.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TC-DIS-002

Test Case Title: Verify that instructors can delete discussion posts

Test Case Description: As an instructor, I want to delete a discussion post to remove inappropriate, disruptive, or off-topic content.

Test Suite: Discussion Moderation

Test Priority: High

Preconditions:

- User is logged in as an instructor
- A discussion post exists

Test Data: A valid discussion post

Test Steps:

1. Navigate to the discussion forum
2. Locate and open the existing discussion post
3. Click on the "Delete" button below the post
4. Confirm the action to delete the post

Postconditions:

- The discussion post no longer appears in the forum

Expected Result: The discussion post is permanently removed from the forum upon instructor's confirmation.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TC-DIS-003

Test Case Title: Verify that instructors can highlight discussion posts

Test Case Description: As an instructor, I want to highlight a discussion post to draw attention to relevant, insightful, or important content.

Test Suite: Discussion Moderation

Test Priority: Medium

Preconditions:

- User is logged in as an instructor
- A discussion post exists

Test Data: A valid discussion post

Test Steps:

1. Navigate to the discussion forum
2. Locate and open the existing discussion post
3. Click on the "Highlight" button below the post
4. Verify that the post is now highlighted in the forum

Postconditions:

- The discussion post is highlighted for other users to easily notice

Expected Result: The selected discussion post is highlighted in the forum, making it more noticeable to other participants.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TC-DIS-004

Test Case Title: Verify that instructors can mute or ban disruptive users

Test Case Description: As an instructor, I want to mute or ban a disruptive user to maintain a positive and respectful learning environment.

Test Suite: Discussion Moderation

Test Priority: High

Preconditions:

- User is logged in as an instructor
- A discussion participant exists

Test Data: A disruptive discussion participant

Test Steps:

1. Navigate to the discussion forum
2. Open the list of participants or users in the forum
3. Identify the disruptive participant
4. Click on the "Mute" or "Ban" button next to the participant's name
5. Provide a valid reason for the action
6. Confirm the action to mute or ban the participant

Postconditions:

- The participant is muted or banned from the discussion forum

Expected Result: The disruptive participant is successfully muted or banned from the forum, and they can no longer post or engage in discussions.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TC-DIS-005

Test Case Title: Verify that moderated actions are logged and reported

Test Case Description: As an instructor, I want to ensure that my moderation actions are accurately logged and reported for transparency and accountability.

Test Suite: Discussion Moderation

Test Priority: High

Preconditions:

- User is logged in as an instructor
- A discussion post or participant exists

Test Data: Moderated discussion posts or participants

Test Steps:

1. Perform a series of moderation actions (e.g., edit, delete, highlight, mute, ban)
2. Navigate to the moderation log
3. Verify that the moderation actions are present in the log

Postconditions:

- Moderation actions are logged and available for review

Expected Result: The moderation actions are accurately logged and reported, allowing instructors to review their previous actions in the forum.

Severity: Major

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 ID: TC-LA-001

Test Case Title: Verify that learning analytics dashboards display engagement metrics

Test Suite: Learning Analytics Dashboard

Test Priority: High

Preconditions:

- User is logged in as an instructor.
- The monitored course has active learners and ongoing engagement.

Test Data: No test data needed

Test Steps:

1. Log in as an instructor.
2. Access the Learning Analytics Dashboard.
3. Select a course with active learners.
4. Review the displayed engagement metrics.

Postconditions:

- Engagement metrics are accurately shown for the selected course.
- No errors or exceptions occur during the process.

Expected Result: Engagement metrics like total learners, weekly active learners, average time spent, and resource access statistics are accurately displayed for the selected course.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-LA-002

Test Case Title: Verify that performance trends are visualized over time

Test Suite: Learning Analytics Dashboard

Test Priority: High

Preconditions:

- User is logged in as an instructor.

- The monitored course has learners who have completed assessments and quizzes.

Test Data: No test data needed

Test Steps:

1. Log in as an instructor.
2. Access the Learning Analytics Dashboard.
3. Select a course with completed assessments and quizzes.
4. Review the performance trends visualization.

Postconditions:

- Performance trends are accurately displayed for the selected course.
- No errors or exceptions occur during the process.

Expected Result: Performance trends like quiz scores, assessment grades, and other performance indicators are visualized over time for the selected course, revealing trends and patterns.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-LA-003

Test Case Title: Verify that instructors can access detailed reports on learner progress

Test Suite: Learning Analytics Dashboard

Test Priority: High

Preconditions:

- User is logged in as an instructor.
- The monitored course has learners with various progress levels.

Test Data: No test data needed

Test Steps:

1. Log in as an instructor.



2. Access the Learning Analytics Dashboard.
3. Select a course with learners at different progress stages.
4. Access the detailed learner progress reports.

Postconditions:

- Detailed learner progress reports are displayed.
- No errors or exceptions occur during the process.

Expected Result: Detailed progress reports show individual learners' completion statuses for modules, lectures, assessments, and quizzes, along with time spent and average grades.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-LA-004

Test Case Title: Verify that analytics data is updated in real-time

Test Suite: Learning Analytics Dashboard

Test Priority: High

Preconditions:

- User is logged in as an instructor.
- The monitored course has active learners and ongoing engagement.

Test Data: No test data needed

Test Steps:

1. Log in as an instructor.
2. Access the Learning Analytics Dashboard.
3. Select a course with active learners.
4. Engage in an activity that affects analytics data, such as a learner completing a lecture or quiz.
5. Monitor the displayed engagement metrics and performance trends.

Postconditions:

- Real-time data updates are reflected in the displayed engagement metrics and performance trends.
- No errors or exceptions occur during the process.

Expected Result: Real-time updates are accurately shown in the displayed engagement metrics and performance trends.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-LA-005

Test Case Title: Verify that instructors can export analytics reports for further analysis

Test Suite: Learning Analytics Dashboard

Test Priority: Medium

Preconditions:

- User is logged in as an instructor.
- The monitored course has learners with completed assessments and quizzes.

Test Data: No test data needed

Test Steps:

1. Log in as an instructor.
2. Access the Learning Analytics Dashboard.
3. Select a course with completed assessments and quizzes.
4. Export the analytics report as a CSV or Excel file.
5. Review the exported data.

Postconditions:

- Analytics report is accurately exported.
- No errors or exceptions occur during the process.

Expected Result: Analytics reports can be exported as CSV or Excel files, retaining the accuracy of the displayed analytics data.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

-----

**Data - 65**

**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 ID: TC-SLF-001

Test Case Title: Verify that learners can create and join groups for projects

Test Case Description: This test case verifies the ability of learners to create new groups for collaborative projects and join existing groups.

Test Suite: Social Learning Features

Test Priority: High

Preconditions:

- Learner is logged into the online learning platform
- The learner has access to the social learning features

Test Data:

- Learner's username and password
- Group project topic or name

Test Steps:

1. Log in as a learner
2. Navigate to the social learning section
3. Select 'Create Group' or 'Join Group' option
4. Enter the required group details (if creating a group)
5. Confirm the creation or selection of the group

Postconditions:

- A new group has been created (if applicable)
- The learner is a member of the selected group (if applicable)

Expected Result:

- The learner can create a new group for a project or join an existing group successfully.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-SLF-002

Test Case Title: Verify that collaboration tools are available for group communication

Test Case Description: This test case verifies the functionality of collaboration tools within groups, enabling learners to effectively communicate and share resources.

Test Suite: Social Learning Features

Test Priority: Medium

Preconditions:

- Learner is logged into the online learning platform
- The learner is a member of a group

Test Data:

- Learner's username and password
- Group project topic or name

Test Steps:

1. Log in as a learner
2. Navigate to the group's page
3. Access the collaboration tools
4. Create a new post or thread
5. Reply to an existing post or thread

Postconditions:

- A new post or thread has been created
- The learner's response is visible within the group

Expected Result:

- Collaboration tools within the group enable learners to create new posts, threads, and engage in discussions with other group members.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-SLF-003

Test Case Title: Verify that learners can share resources and feedback within group

Test Case Description: This test case validates the functionality of resource sharing and feedback mechanisms within groups, allowing learners to exchange information and provide constructive criticism.

Test Suite: Social Learning Features

Test Priority: Medium

Preconditions:

- Learner is logged into the online learning platform
- The learner is a member of a group
- There is a resource available for sharing

Test Data:

- Learner's username and password
- Group project topic or name
- Resource file or link

Test Steps:

1. Log in as a learner
2. Navigate to the group's page
3. Access the resource sharing feature
4. Upload or share the resource
5. Provide feedback on a shared resource

Postconditions:

- The resource is visible within the group
- The feedback is associated with the shared resource

Expected Result:

- Learners can share resources and provide feedback within groups, enabling better collaboration and knowledge exchange.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-SLF-004

Test Case Title: Verify that instructors can assign group projects and monitor progress

Test Case Description: This test case verifies the ability of instructors to assign group projects and monitor the progress of learners within the groups.

Test Suite: Social Learning Features

Test Priority: High

Preconditions:

- Instructor is logged into the online learning platform
- Instructor has access to the social learning features
- At least one group has been created
- Learners are members of the group

Test Data:

- Instructor's username and password
- Group project details

Test Steps:

1. Log in as an instructor

2. Navigate to the social learning section
3. Access the group management feature
4. Assign a group project
5. Monitor the progress of the group

Postconditions:

- The group project has been assigned
- The instructor can view the progress of the group members

Expected Result:

- Instructors can assign group projects and monitor the progress of learners effectively.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-SLF-005

Test Case Title: Verify that social sharing features allow learners to share achievements

Test Case Description: This test case checks the social sharing functionalities, ensuring learners can share their achievements outside the platform.

Test Suite: Social Learning Features

Test Priority: Medium

Preconditions:

- Learner is logged into the online learning platform
- The learner has completed a course or received a certification

Test Data:

- Learner's username and password
- Certification or course completion information

Test Steps:



1. Log in as a learner
2. Navigate to the certifications or courses section
3. Locate the completed course or certification
4. Use the social sharing feature to share the achievement
5. Verify the shared post on a supported social media platform

Postconditions:

- The achievement has been shared on a supported social media platform

Expected Result:

- Learners can share their achievements on supported social media platforms using the social sharing features.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

-----

**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 is able to 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
- User is on the course creation page

Test Data: Prepared media files (images, audio, video)

Test Steps:

1. Log in as an instructor
2. Navigate to the course creation page
3. Click on "Add Multimedia Presentation"
4. Upload media files as required

5. Arrange media files in the desired order

6. Save the presentation

Postconditions:

- Multimedia presentation is saved and viewable in the course outline

Expected Result:

- A multimedia presentation is successfully created and saved, with the ability to view and edit it later.

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 the course content.

Test Suite: Content Authoring Tools

Test Priority: High

Preconditions:

- User is logged in as an instructor
- User is on the course creation page

Test Data: Pre-built interactive module (quiz, survey, or discussion)

Test Steps:

1. Log in as an instructor
2. Navigate to the course creation page
3. Click on "Add Interactive Module"
4. Select a pre-built interactive module or create a new one
5. Configure the settings and options for the module

6. Save the module

7. Integrate the module into the course outline

Postconditions:

- Interactive module is saved, configured, and integrated into the course

Expected Result:

- An interactive module is successfully created, configured, and integrated into the course, allowing students to participate and complete the assigned task.

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 are supported

Test Case Description: This test case verifies that the content authoring tools support rich media resources such as videos and animations, ensuring proper rendering and functionality.

Test Suite: Content Authoring Tools

Test Priority: High

Preconditions:

- User is logged in as an instructor
- User is on the course creation page

Test Data: Video or animation file

Test Steps:

1. Log in as an instructor
2. Navigate to the course creation page
3. Click on "Add Rich Media Resource"
4. Upload the video or animation file

5. Save the resource and preview it in the course outline

Postconditions:

- Rich media resource is saved and renders properly

Expected Result:

- A rich media resource, such as a video or animation, can be successfully uploaded, saved, and rendered within the course outline.

Severity: Major

Priority: High

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 are user-friendly, intuitive, and easy for instructors to use.

Test Suite: Content Authoring Tools

Test Priority: Medium

Preconditions:

- User is logged in as an instructor
- User is on the course creation page

Test Steps:

1. Log in as an instructor
2. Navigate to the course creation page
3. Explore the different features and options of the content authoring tools
4. Create a simple course, including multimedia presentations, interactive modules, and rich media resources
5. Save and preview the course

Postconditions:

- Instructor has successfully created a course using the content authoring tools

Expected Result:

- The content authoring tools are user-friendly and intuitive, enabling an instructor to create a course with various learning materials without difficulty.

Severity: Minor

Priority: Medium

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 can preview and edit course content, including multimedia presentations, interactive modules, and rich media resources, before publishing the course.

Test Suite: Content Authoring Tools

Test Priority: High

Preconditions:

- User is logged in as an instructor
- User has created a course with multimedia presentations, interactive modules, and rich media resources

Test Steps:

1. Log in as an instructor
2. Navigate to the course creation page
3. Preview the course and its components
4. Make changes to any part of the course content as needed
5. Save the changes
6. Preview the course again to ensure that the changes were applied as expected

Postconditions:

- Instructor has previewed and made changes to the course content

Expected Result:

- The course content is accurately reflected in the preview mode, and instructors are able to make any necessary edits before publishing the course.

---

**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: Learning Path Sequential Courses and Modules

Test Case ID: TC-OLP-LPS-001

Test Case Title: Verify that learning paths include sequential courses and modules

Test Case Description: Validate that the learning paths are structured according to their respective courses and modules, ensuring a logical progression.

Test Suite: Learning Paths

Test Priority: High

Preconditions:

1. User is logged into the platform.
2. User has access to a learning path.

Test Data: Learning path with at least two courses and two modules

Test Steps:

1. Navigate to the learning path.
2. Verify that the first course is displayed.
3. Enter the first course and verify that the first module is displayed.
4. Move to the next module and verify that it is the second module in the course.
5. Exit the course and verify that the second course is displayed.

Postconditions: Learning path progress is saved and displayed correctly.

Expected Result: The learning path displays courses and modules sequentially based on their alignment.

Severity: Major

Priority: High

Type of Testing: System Testing, Functional Testing

Test Case Approach: Positive

### Test Case 2: Curriculum Learning Objectives



Test Case ID: TC-OLP-LPS-002

Test Case Title: Verify that curricula are aligned with specific learning objectives

Test Case Description: Confirm that the curricula clearly outline specific learning objectives and that the courses within the curricula support these objectives.

Test Suite: Curriculum

Test Priority: High

Preconditions:

1. User is logged into the platform.
2. User has access to a curriculum.

Test Data: Curriculum with at least two learning objectives and two courses

Test Steps:

1. Navigate to the curriculum.
2. Verify that specific learning objectives are displayed.
3. Select the first course and verify that its content is relevant to the first learning objective.
4. Select the second course and verify that its content is relevant to the second learning objective.

Postconditions: Learning objectives and course content are consistent.

Expected Result: Curricula are aligned with specific learning objectives, and courses within the curricula directly support these objectives.

Severity: Major

Priority: High

Type of Testing: System Testing, Functional Testing

Test Case Approach: Positive

Test Case 3: Progress Tracking

Test Case ID: TC-OLP-LPS-003

Test Case Title: Verify that learners can track their progress within a learning path

Test Case Description: Validate that learners can view their progress through a learning path, including the

percentage completed and courses and modules remaining.

Test Suite: Learning Paths

Test Priority: Medium

Preconditions:

1. User is logged into the platform.
2. User has access to a learning path with progress.

Test Data: Learning path with completed and incomplete courses and modules

Test Steps:

1. Navigate to the learning path.
2. Verify that progress is displayed as a percentage.
3. Check that completed courses and modules are marked as such.
4. Verify that the number of remaining courses and modules is accurate.

Postconditions: Progress tracking information is displayed and updated.

Expected Result: Learners can view and track their progress through a learning path.

Severity: Major

Priority: Medium

Type of Testing: System Testing, Functional Testing

Test Case Approach: Positive

Test Case 4: Customizable Learning Paths

Test Case ID: TC-OLP-LPS-004

Test Case Title: Verify that instructors can customize learning paths for different needs

Test Case Description: Confirm that instructors can modify learning paths by reordering courses and modules or by adding and removing content.

Test Suite: Learning Path Builder

Test Priority: Medium

Preconditions:

1. Instructor is logged into the platform.
2. Instructor has access to a learning path.

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. Modify the learning path by reordering the courses or modules.
4. Add or remove courses or modules within the learning path.
5. Save the changes and verify that the learning path reflects the modifications.

Postconditions: Modified learning path is saved and accessible.

Expected Result: Instructors can customize learning paths to suit various needs.

Severity: Major

Priority: Medium

Type of Testing: System Testing, Functional Testing

Test Case Approach: Positive

Test Case 5: Notifications for Upcoming Courses or Milestones

Test Case ID: TC-OLP-LPS-005

Test Case Title: Verify that notifications are sent for upcoming courses or milestones

Test Case Description: Validate that learners receive notifications for upcoming courses or milestones within a learning path according to the specified notification settings.

Test Suite: Notification System

Test Priority: Low

Preconditions:

1. User is logged into the platform.
2. User has access to a learning path.
3. Notification settings are enabled.

Test Data: Learning path with upcoming courses or milestones

Test Steps:

1. Log in as a learner.
2. Ensure that notification settings are enabled.
3. Wait for the specified period before the upcoming course or milestone.
4. Verify that a notification regarding the upcoming course or milestone is received.

Postconditions: Notifications for upcoming courses or milestones are sent.

Expected Result: Learners receive notifications for upcoming courses or milestones within a learning path.

Severity: Major

Priority: Low

Type of Testing: System 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-001

Test Case Title: Verify that attendance is automatically recorded during live sessions

Test Suite: Attendance Tracking

Test Priority: High

Preconditions:

- Instructor has set up a live session for a course
- Learner has joined the live session

Test Data: No test data needed

Test Steps:

1. Log in as an instructor and create a live session for a course.
2. Invite learners to join the live session.
3. Start the live session and ensure it is running smoothly.
4. Monitor learner participation during the live session.
5. End the live session.

Postconditions:

- Attendance records are created for all participants
- Records include start and end times for each learner

Expected Result:

- Accurate attendance records are automatically created and saved for each learner

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC-002

Test Case Title: Verify that instructors can view attendance reports for each session

Test Suite: Attendance Tracking

Test Priority: High

Preconditions:

- Instructor has set up and conducted a live session

Test Data: No test data needed

Test Steps:

1. Log in as an instructor.
2. Navigate to the Reports section of the platform.
3. Select the attendance report for the specific session.
4. Review the attendance records for all participants.

Postconditions:

- Instructor can access and view the attendance report

Expected Result:

- Instructor can view an accurate and detailed attendance report with names, start/end times, and other relevant data

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC-003

Test Case Title: Verify that learners can see their attendance records

Test Suite: Attendance Tracking

Test Priority: Medium

Preconditions:

- Learner has joined and participated in a live session

Test Data: No test data needed

Test Steps:

1. Log in as a learner.
2. Navigate to the My Progress section of the platform.
3. View the attendance records for all sessions.
4. Check the accuracy and completeness of the attendance records for each session.

Postconditions:

- Learner can access and view their attendance records

Expected Result:

- Learner can view their attendance records, including start/end times and other relevant data

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC-004

Test Case Title: Verify that attendance data is integrated with participation metrics

Test Suite: Attendance Tracking

Test Priority: Medium

Preconditions:

- Instructor has set up and conducted a live session
- Learners have joined and participated in the session

Test Data: No test data needed

Test Steps:

1. Log in as an instructor.
2. Navigate to the Reports section of the platform.
3. Select the participation metric report for the specific session.
4. Review the attendance records and other participation metrics for all participants.

Postconditions:

- Attendance records are integrated with other participation metrics (e.g., time spent, questions asked)

Expected Result:

- Accurate attendance records and participation metrics are available in a single report

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC-005

Test Case Title: Verify that notifications are sent for missed sessions



Test Suite: Attendance Tracking

Test Priority: Medium

Preconditions:

- Instructor has set up and conducted a live session
- Learners are registered for the session but do not attend

Test Data: No test data needed

Test Steps:

1. Log in as an instructor.
2. Navigate to the Notifications section of the platform.
3. Check for notifications of missed sessions for registered learners.

Postconditions:

- Notifications are sent for missed sessions

Expected Result:

- Accurate notifications are sent to learners who missed the session

Severity: Major

Priority: Medium

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 ID: TC001

Test Case Title: Verify that content licenses are assigned to each resource

Test Case Description: This test case verifies that content licenses are properly assigned to individual resources within the online learning platform.

Test Suite: Content Management

Test Priority: High

Preconditions:

- User has logged in with content creator permissions
- A new resource has been added to the platform

Test Data: Newly added resource

Test Steps:

1. Log in as a content creator
2. Add a new resource to the platform
3. Assign a content license to the newly added resource
4. Save and publish the resource

Postconditions:

- The resource is published with the correct content license

Expected Result: The resource is successfully published with the assigned content license.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that usage rights and permissions are clearly defined

Test Case Description: This test case ensures that the platform clearly defines usage rights and permissions for all content resources.

Test Suite: Content Management

Test Priority: High

Preconditions:

- User has logged in with content creator permissions
- A content resource with a content license is available on the platform

Test Data: Content resource with a content license

Test Steps:

1. Log in as a content creator
2. Access the content resource information page
3. Review the usage rights and permissions associated with the content license

Postconditions:

- Usage rights and permissions are clearly displayed

Expected Result: The usage rights and permissions are clearly defined and displayed for the content resource.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that content creators can set restrictions on usage and distribution

Test Case Description: This test case checks if content creators can set restrictions on usage and distribution for their resources.

Test Suite: Content Management

Test Priority: High

Preconditions:

- User has logged in with content creator permissions
- A new resource has been added to the platform

Test Data: Newly added resource

Test Steps:

1. Log in as a content creator
2. Add a new resource to the platform
3. Set restrictions on usage and distribution for the new resource
4. Save and publish the resource

Postconditions:

- The resource is published with the specified restrictions

Expected Result: The resource is successfully published with the specified restrictions on usage and distribution.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that instructors and learners are informed of content licensing terms

Test Case Description: This test case confirms that instructors and learners are informed of content licensing terms when accessing resources.

Test Suite: User Interface

Test Priority: Medium

Preconditions:

- User has logged in with instructor or learner permissions
- A content resource with a content license is available on the platform

Test Data: Content resource with a content license

Test Steps:

1. Log in as an instructor or learner
2. Access the content resource information page
3. Review the displayed content licensing terms

Postconditions:

- Content licensing terms are clearly visible

Expected Result: Content licensing terms are clearly displayed and visible to instructors and learners when accessing resources.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that notifications are sent for license expiration or renewal

Test Case Description: This test case verifies that the platform sends notifications for license expiration or renewal to content creators.

Test Suite: Notification System

Test Priority: Medium

Preconditions:

- User has logged in with content creator permissions
- A content resource with a content license is available on the platform
- The content license has an expiration date set

Test Data: Content resource with a content license nearing expiration

Test Steps:

1. Log in as a content creator
2. Access the content resource with the nearing expiration date
3. Wait for the expiration date to approach

Postconditions:

- A notification is received regarding the license expiration

Expected Result: The content creator receives a notification about the license expiration or renewal for the content resource.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Time-based Negative

-----

**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 ID: TC001

Test Case Title: Verify that instructors can create and save multiple versions of content.

Test Case Description: As an instructor, I should be able to create different versions of my course material and save them for future reference.

Test Suite: Content Versioning

Test Priority: High

Preconditions:

- Log in as an instructor.
- Navigate to the course editing page.

Test Data: Any course material (e.g., text, images, videos)

Test Steps:

1. Edit the course material.
2. Click on the "Create New Version" button.
3. Save the new version with a unique name.
4. Repeat steps 1-3 to create additional versions.

Postconditions:

- Multiple versions of the course material are saved and accessible.
- The latest version is set as the default.

Expected Result: The system allows instructors to create and save multiple unique versions of their course materials.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that version history is maintained and accessible.

Test Case Description: Instructors should be able to access the version history of their course materials and view details of each version.

Test Suite: Content Versioning

Test Priority: High

Preconditions:



- Log in as an instructor.
- Create and save at least two versions of course material.

Test Data: N/A

Test Steps:

1. Navigate to the course editing page.
2. Locate and open the course material version dropdown.
3. Select a previous version from the list.
4. View the details of the selected version.

Postconditions:

- The selected version's details are displayed accurately.

Expected Result: Instructors can access and view the version history, including details of each version, for their course materials.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that instructors can compare different versions of content.

Test Case Description: Instructors should have the ability to compare two versions of their course materials side-by-side for easy identification of changes.

Test Suite: Content Versioning

Test Priority: Medium

Preconditions:

- Log in as an instructor.
- Create and save at least two versions of course material.

Test Data: N/A

Test Steps:

1. Navigate to the course editing page.
2. Locate and open the course material version dropdown.
3. Select two versions to compare from the list.
4. View the side-by-side comparison of the selected versions.

Postconditions:

- The differences between the two versions are highlighted.

Expected Result: Instructors can compare different versions of their course materials and identify the changes made between them.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that previous versions can be restored if needed.

Test Case Description: Instructors should be able to restore a previous version of their course materials as the default version if required.

Test Suite: Content Versioning

Test Priority: Medium

Preconditions:

- Log in as an instructor.
- Create and save at least two versions of course material.

Test Data: N/A

Test Steps:

1. Navigate to the course editing page.
2. Locate and open the course material version dropdown.

3. Select a previous version from the list.

4. Click the "Restore" button to set the selected version as the default.

Postconditions:

- The selected version is now the default version.

Expected Result: Instructors can restore a previous version of their course materials as the default version if needed.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that notifications are sent for significant content updates.

Test Case Description: The system should send notifications to students when significant updates are made to the course materials.

Test Suite: Content Versioning

Test Priority: Low

Preconditions:

- Log in as an instructor.
- Create and save at least two versions of course material.
- Have students enrolled in the course.

Test Data: N/A

Test Steps:

1. Make a significant update to the course material.
2. Save the update as a new version.

Expected Result: Students receive a notification about the significant update to the course materials.

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 ID: TC-API-001

Test Case Title: Verify that APIs are available for key platform functionalities

Test Case Description: This test case verifies the availability of APIs for crucial platform features, ensuring proper integration with third-party applications.

Test Suite: API Integration

Test Priority: High

Preconditions:

- Online Learning Platform is installed and running
- Third-party application is installed and ready to connect

Test Data: No test data needed

Test Steps:

1. Access the Online Learning Platform API documentation
2. Identify the key platform functionalities
3. Check for corresponding API endpoints in the documentation
4. Attempt to connect to the identified API endpoints using the third-party application
5. Verify successful connection and data retrieval

Postconditions:

- API connections are functioning as expected

Expected Result: All key platform functionalities have corresponding API endpoints, and third-party applications can successfully connect and retrieve data from these endpoints.

Severity: Major

Priority: High

Type of Testing: API Testing, Functional Testing

Test Case Approach: Positive

Test Case ID: TC-API-002

Test Case Title: Verify that comprehensive API documentation is provided

Test Case Description: This test case checks if the API documentation covers all necessary aspects for developers to understand and interact with the platform's APIs.

Test Suite: API Integration

Test Priority: High

Preconditions:

- Online Learning Platform is installed and running

Test Data: No test data needed

Test Steps:

1. Access the Online Learning Platform API documentation
2. Verify that all endpoints and their respective methods are documented
3. Check if request parameters and response formats are explained
4. Ensure that any authentication or authorization requirements are outlined
5. Validate that error handling and troubleshooting information is provided

Postconditions:

- API documentation is complete and accurate

Expected Result: The API documentation is comprehensive and covers all necessary aspects, enabling developers to understand and interact with the platform's APIs without encountering gaps or ambiguities.

Severity: Major

Priority: High

Type of Testing: API Testing, Documentation Testing, Functional Testing

Test Case Approach: Positive

Test Case ID: TC-API-003

Test Case Title: Verify that secure access protocols are implemented

Test Case Description: This test case ensures the proper implementation of secure access protocols for the platform's APIs.

Test Suite: API Integration

Test Priority: High

Preconditions:

- Online Learning Platform is installed and running
- Third-party application is installed and ready to connect

Test Data: No test data needed

Test Steps:

1. Access the Online Learning Platform API documentation
2. Identify any authentication or authorization requirements
3. Attempt to connect to API endpoints using invalid credentials or unauthorized access
4. Verify that the platform rejects unauthorized access attempts
5. Check if the platform supports secure data transmission protocols such as HTTPS

Postconditions:

- Secure access protocols are functioning as expected

Expected Result: The platform enforces secure access protocols, rejecting unauthorized access attempts and supporting secure data transmission protocols.

Severity: Critical

Priority: High

Type of Testing: API Testing, Security Testing, Functional Testing

Test Case Approach: Negative

Test Case ID: TC-API-004

Test Case Title: Verify that rate limiting and monitoring features are in place

Test Case Description: This test case validates rate limiting and monitoring features to prevent abuse and ensure stable platform performance.

Test Suite: API Integration

Test Priority: Medium

Preconditions:

- Online Learning Platform is installed and running
- Third-party application is installed and ready to connect

Test Data: No test data needed

Test Steps:

1. Access the Online Learning Platform API documentation
2. Identify any mention of rate limiting or API usage monitoring
3. Send multiple requests to an API endpoint within a short period
4. Verify that the platform responds with appropriate error messages or throttling mechanisms
5. Confirm that the platform logs abnormal usage patterns or suspicious behavior

Postconditions:

- Rate limiting and monitoring features are functioning as expected

Expected Result: The platform enforces rate limiting and monitors API usage to maintain stability and prevent abuse.

Severity: Major

Priority: Medium

Type of Testing: API Testing, Performance Testing, Non-Functional Testing, Security Testing

Test Case Approach: Negative

Test Case ID: TC-API-005

Test Case Title: Verify that third-party applications can authenticate and access APIs

Test Case Description: This test case checks if third-party applications can authenticate and access the platform's APIs.

Test Suite: API Integration

Test Priority: High

Preconditions:

- Online Learning Platform is installed and running
- Third-party application is installed and ready to connect



Test Data: Test application credentials for API authentication

Test Steps:

1. Obtain valid credentials for third-party application authentication
2. Attempt to authenticate with the Online Learning Platform API using the test credentials
3. Access the API endpoint for a key platform functionality
4. Verify successful data exchange between the third-party application and the platform's API

Postconditions:

- Third-party applications can authenticate and access APIs as expected

Expected Result: Third-party applications can authenticate using valid credentials and successfully access and interact with the platform's APIs.

Severity: Major

Priority: High

Type of Testing: API Testing, Integration Testing, Functional Testing

Test Case Approach: Positive

-----

**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 ID: TC001

Test Case Title: Verify that data visualization tools are available for analysis

Test Case Description: As an administrator, verify the presence and proper function of data visualization tools within the online learning platform to analyze learner performance.

Test Suite: Data Analytics and Insights

Test Priority: High

Preconditions:

- Online learning platform account with administrator privileges
- At least one course with multiple learners and performance data

Test Data: No test data needed

Test Steps:

1. Log in as an administrator
2. Navigate to the data analytics and insights section
3. Select a course with multiple learners and performance data
4. Verify the presence of data visualization tools (charts, graphs, etc.)

5. Interact with data visualization tools to ensure proper function

Postconditions:

- Data visualization tools are available and functional for administrators

Expected Result: Administrators can access and interact with data visualization tools within the online learning platform to analyze learner performance.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC002

Test Case Title: Verify that reports can be generated for learner performance metrics

Test Case Description: As an administrator, verify the ability to generate reports on various learner performance metrics within the online learning platform.

Test Suite: Data Analytics and Insights

Test Priority: High

Preconditions:

- Online learning platform account with administrator privileges
- At least one course with multiple learners and performance data

Test Data: No test data needed

Test Steps:

1. Log in as an administrator
2. Navigate to the data analytics and insights section
3. Select a course with multiple learners and performance data
4. Access the report generation feature

5. Verify the availability of various learner performance metrics

6. Generate reports for selected learner performance metrics

Postconditions:

- Reports on various learner performance metrics are available for administrators

Expected Result: Administrators can generate reports on various learner performance metrics within the online learning platform.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC003

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 are available and provide accurate insights into learner engagement and outcomes.

Test Suite: Data Analytics and Insights

Test Priority: High

Preconditions:

- Online learning platform account with administrator privileges
- At least one course with multiple learners actively participating

Test Data: No test data needed

Test Steps:

1. Log in as an administrator
2. Navigate to the data analytics and insights section
3. Select a course with multiple learners actively participating

4. Access real-time analytics for the selected course

5. Verify the accuracy and relevance of the real-time engagement and outcome data

Postconditions:

- Real-time analytics are available and provide accurate insights into learner engagement and outcomes

Expected Result: Administrators can access real-time analytics that accurately reflect learner engagement and outcome data.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC004

Test Case Title: Verify that data can be exported for further analysis

Test Case Description: As an administrator, verify the ability to export data for further analysis outside the online learning platform.

Test Suite: Data Analytics and Insights

Test Priority: Medium

Preconditions:

- Online learning platform account with administrator privileges
- At least one course with multiple learners and performance data

Test Data: No test data needed

Test Steps:

1. Log in as an administrator
2. Navigate to the data analytics and insights section
3. Select a course with multiple learners and performance data

4. Access the data export feature
5. Verify the available export formats (CSV, Excel, etc.)
6. Export data in a selected format

Postconditions:

- Data can be exported in various formats for further analysis

Expected Result: Administrators can export data in various formats for further analysis outside the online learning platform.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC005

Test Case Title: Verify that notifications are sent for significant data trends or anomalies

Test Case Description: As an administrator, verify that notifications for significant data trends or anomalies are properly configured and sent.

Test Suite: Data Analytics and Insights

Test Priority: Medium

Preconditions:

- Online learning platform account with administrator privileges
- Notification settings for data trends and anomalies are configured
- At least one course with learners actively participating

Test Data: No test data needed

Test Steps:

1. Log in as an administrator

2. Configure notification settings for data trends and anomalies
3. Navigate to the data analytics and insights section
4. Select a course with learners actively participating
5. Trigger a significant data trend or anomaly
6. Wait for the configured notification to be received

Postconditions:

- Notifications are properly sent for significant data trends or anomalies

Expected Result: Administrators receive notifications for significant data trends or anomalies within the online learning platform.

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 ID: TC001

Test Case Title: Verify that administrators can create and assign user roles

Test Case Description: As an administrator, verify the ability to create a new user role and assign it to a user account.

Test Suite: User Roles and Permissions

Test Priority: High

Preconditions:

- Login as an administrator
- Access the user roles management page

Test Data:

- New user role name: "Content Manager"
- Existing user account to assign the new role

Test Steps:

1. Navigate to the user roles management page.
2. Click on the "Create Role" button.



3. Enter the new role name "Content Manager" and save the changes.
4. Locate the newly created role in the list and click on the "Assign Users" button.
5. Search for an existing user account and select it.
6. Click on the "Assign" button to assign the user to the new role.

Postconditions:

- The new user role "Content Manager" is visible in the roles list.
- The assigned user is listed under the new role.

Expected Result:

- The test verifies that administrators can create and assign new user roles.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that permissions are customizable for each role

Test Case Title: Verify that permissions are customizable for each role

Test Case Description: As an administrator, verify the ability to enable or disable specific permissions for a user role.

Test Suite: User Roles and Permissions

Test Priority: High

Preconditions:

- Login as an administrator
- Access the user roles management page

Test Data:

- Existing user role: "Instructor"

Test Steps:

1. Navigate to the user roles management page.
2. Locate the "Instructor" role in the list and click on the "Edit" button.
3. Scroll down to the permissions section.
4. Enable or disable specific permissions, such as "Create Quiz" or "Delete Lecture".
5. Save the changes.

Postconditions:

- The "Instructor" role has the updated customized permissions.

Expected Result:

- The test verifies that permissions are customizable for each role.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users are restricted to functionalities based on their roles

Test Case Title: Verify that users are restricted to functionalities based on their roles

Test Case Description: As a user with a specific role, verify the inability to access features and functionalities restricted to other roles.

Test Suite: User Roles and Permissions

Test Priority: High

Preconditions:

- Login as a user with a specific role (e.g., Student)

Test Data:

- Restricted feature: Access to the course creation page

Test Steps:

1. Log in as a user with the "Student" role.

2. Navigate to the course creation page.

3. Attempt to create a new course.

Postconditions:

- The user is unable to create a new course.

Expected Result:

- The test verifies that users are restricted to functionalities based on their roles.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Negative

Test Case ID: TC004

Test Case Title: Verify that changes to roles and permissions are logged

Test Case Title: Verify that changes to roles and permissions are logged

Test Case Description: As an administrator, verify that changes made to user roles and permissions are recorded in the system logs.

Test Suite: User Roles and Permissions

Test Priority: Medium

Preconditions:

- Login as an administrator
- Access the user roles management page

Test Steps:

1. Navigate to the user roles management page.
2. Create a new user role or modify an existing one.
3. Save the changes.
4. Navigate to the system logs.

Postconditions:

- The system logs contain an entry for the role or permission change.

Expected Result:

- The test verifies that changes to roles and permissions are logged.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that users are notified of their assigned roles and permissions

Test Case Title: Verify that users are notified of their assigned roles and permissions

Test Case Description: As a user, verify that a notification is received upon assignment or modification of a user role.

Test Suite: User Roles and Permissions

Test Priority: Medium

Preconditions:

- Login as an administrator
- Access the user roles management page

Test Data:

- Existing user account for notification testing

Test Steps:

1. Navigate to the user roles management page.
2. Assign or modify a user role for the selected user account.
3. Save the changes.

Postconditions:

- The user receives a notification about the role change.

Expected Result:

- The test verifies that users are notified of their assigned roles and permissions.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

-----

## **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 provide a star rating for courses and instructors to reflect my satisfaction level.

Test Suite: User Feedback and Rating System

Test Priority: High

Preconditions:

- Learner must be logged in
- The learner has completed at least one lecture

Test Data: Course ID, Instructor ID, Learner's rating (1-5 stars)

Test Steps:

1. Log in as a learner
2. Access the course or instructor profile page
3. Locate the rating section
4. Enter a rating between 1-5 stars for the course or instructor
5. Confirm and submit the rating

Postconditions:

- The rating is displayed on the course or instructor profile page

Expected Result: The system allows learners to rate courses and instructors on a scale of 1-5 stars, with the entered rating displayed publicly.

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 the course or instructor using a feedback form.

Test Suite: User Feedback and Rating System

Test Priority: Medium

Preconditions:

- Learner must be logged in
- The learner has completed at least one lecture

Test Data: Course ID, Instructor ID, Learner's feedback text (minimum 50 characters)

Test Steps:

1. Log in as a learner
2. Access the course or instructor profile page
3. Locate the feedback form
4. Enter at least 50 characters of feedback for the course or instructor in the provided text field
5. Submit the feedback form

Postconditions:

- The feedback is saved and associated with the appropriate course or instructor

Expected Result: The system provides a feedback form for learners to submit detailed comments on courses or instructors.

Severity: Major

Priority: Medium

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 reply to feedback from learners to engage with them and improve the learning experience.

Test Suite: User Feedback and Rating System

Test Priority: High

Preconditions:

- Instructor must be logged in
- Learners have provided feedback on the instructor's course or profile

Test Data: Instructor's account, Learner's feedback

Test Steps:

1. Log in as an instructor
2. Access the feedback or course evaluation page
3. View the learners' feedback
4. Compose and submit a response

Postconditions:

- The response is saved and displayed along with the original feedback

Expected Result: Instructors can view and respond to learners' feedback on courses and their profiles.

Severity: Major

Priority: High

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 learner, I want to view the overall rating or popularity of a course or instructor before enrollment.

Test Suite: User Feedback and Rating System



Test Priority: High

Preconditions:

- Multiple learners have rated courses or instructors

Test Data: Course ID, Instructor ID, Existing ratings (minimum 3 learners)

Test Steps:

1. Access the course or instructor profile page
2. Locate the aggregated ratings section
3. Verify the total number of ratings and the average rating displayed

Postconditions:

- The aggregated ratings are correctly calculated and displayed

Expected Result: The system calculates the average rating of a course or instructor based on all the individual ratings provided by learners.

Severity: Major

Priority: High

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 for honest and open communication.

Test Suite: User Feedback and Rating System

Test Priority: Medium

Preconditions:

- Learner must be logged in
- The learner has completed at least one lecture

Test Data: Course ID, Instructor ID, Learner's feedback text (minimum 50 characters)

Test Steps:

1. Log in as a learner
2. Access the course or instructor profile page
3. Locate the feedback form
4. Select the anonymous feedback option before submitting the feedback
5. Submit the feedback form

Postconditions:

- The feedback is saved and associated with the appropriate course or instructor, but the learner's identity is not revealed

Expected Result: The system provides an option for learners to submit anonymous feedback on courses or instructors.

Severity: Minor

Priority: Medium

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 ID: TC001

Test Case Title: Verify that learners can connect their social media accounts to the platform

Test Case Description: As a learner, I want to link my social media accounts (Facebook, Twitter, LinkedIn) to my online learning platform account so that I can easily share my achievements and course completions.

Test Suite: Social Media Integration

Test Priority: High

Preconditions:

- Learner has a valid account on the online learning platform
- Learner is logged in to their account
- The online learning platform has Facebook, Twitter, and LinkedIn integration enabled

Test Data: Learner's credentials for social media accounts

Test Steps:

1. Log in to the online learning platform as a learner

2. Navigate to the account settings page
3. Click on the "Connect social media accounts" button
4. Select the social media platform you want to connect (Facebook, Twitter, LinkedIn)
5. Enter your credentials for the chosen social media platform and allow access
6. Verify that the social media account is now connected

Postconditions:

- The selected social media account is connected, allowing learners to share content directly

Expected Result: Connected social media accounts are displayed in the learner's account settings, enabling learners to share achievements and course completions

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that achievement and course completion notifications can be shared on social media

Test Case Description: Learners should be able to share their achievements, course completions, and certificates on their connected social media accounts to showcase their progress and connect with peers.

Test Suite: Social Media Integration

Test Priority: High

Preconditions:

- Learner has a valid account on the online learning platform
- Learner is logged in to their account
- The learner has connected at least one social media account (Facebook, Twitter, LinkedIn)
- The learner has completed a course and earned an achievement or certificate

Test Data: Learner's course completion and achievement data

Test Steps:

1. Log in to the online learning platform as a learner
2. Navigate to the My Courses page
3. Select a completed course for which you have received an achievement or certificate
4. Click the "Share on social media" button
5. Choose the connected social media account where you want to share the achievement
6. In the post preview, verify that the shared content (achievement badge/certificate, course title, and a link to the course) appears correctly
7. Click the "Post" button to share the achievement on the chosen social media platform
8. Verify that the achievement has been published on the social media platform

Postconditions:

- Achievement or course completion is shared on the chosen social media platform

Expected Result: Achievements, course completions, and certificates can be successfully shared on the chosen social media platform with correct and complete information

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that learners can control the privacy settings of their shared content

Test Case Description: Learners should have the ability to configure the visibility of their shared achievements and course completions based on their privacy preferences.

Test Suite: Social Media Integration

Test Priority: High

Preconditions:

- Learner has a valid account on the online learning platform
- Learner is logged in to their account

- The learner has connected at least one social media account (Facebook, Twitter, LinkedIn)
- The learner has completed a course and earned an achievement or certificate

Test Data: Learner's course completion and achievement data

Test Steps:

1. Log in to the online learning platform as a learner
2. Navigate to the My Courses page
3. Select a completed course for which you have received an achievement or certificate
4. Click the "Share on social media" button
5. Choose the connected social media account where you want to share the achievement
6. Modify the visibility settings (e.g., public, friends, only me)
7. In the post preview, verify that the chosen visibility settings are applied
8. Click the "Post" button to share the achievement on the chosen social media platform
9. Verify that the achievement appears on the social media platform with the chosen privacy settings applied

Postconditions:

- Shared achievements or course completions are visible according to the learner's preferred privacy settings

Expected Result: Learners can successfully share achievements or course completions with customizable privacy settings on their preferred social media platform

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that instructors can promote courses and updates through social media

Test Case Description: Instructors should be able to share course updates, new lectures, and announcements on social media platforms to reach a wider audience and engage with learners.

Test Suite: Social Media Integration

Test Priority: Medium

Preconditions:

- Instructor has a valid account on the online learning platform
- Instructor is logged in to their account
- The instructor has connected at least one social media account (Facebook, Twitter, LinkedIn)
- The instructor has created a course with lectures and updates

Test Data: Instructor's course updates, new lectures, and announcements

Test Steps:

1. Log in to the online learning platform as an instructor
2. Navigate to the course management page
3. Click on the "Promote on social media" button
4. Choose the social media platform where you want to promote the course
5. In the post preview, verify that the shared content (e.g., course title, image, a link to the course, and a short description) appears correctly
6. Click the "Post" button to share the course promotion on the chosen social media platform
7. Verify that the course promotion appears on the social media platform

Postconditions:

- Course updates, new lectures, and announcements can be shared on the chosen social media platform

Expected Result: Instructors can promote their courses with correct and complete information on the chosen social media platform

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that analytics track the reach and engagement of shared content

Test Case Description: Platform administrators should be able to view engagement metrics (e.g., likes, shares, comments) and reach (e.g., impressions, clicks) for shared achievements, course completions, and instructor updates on social media platforms.

Test Suite: Social Media Integration

Test Priority: Low

Preconditions:

- Learner or instructor has a valid account on the online learning platform
- Learner or instructor is logged in to their account
- Learner or instructor has connected at least one social media account (Facebook, Twitter, LinkedIn)
- Learner or instructor has shared achievements, course completions, or instructor updates on social media

Test Data: Shared content engagement metrics and reach data on social media platforms

Test Steps:

1. Log in to the online learning platform as an administrator
2. Navigate to the social media analytics page
3. Verify that the analytics page displays the number of shares, likes, comments, and impressions for the shared content
4. Click on each entry (share, like, comment, impression) to view the related details (e.g., user, platform, and timestamp)
5. Verify that the analytics data matches the engagement metrics and reach data on the respective social media platforms

Postconditions:

- Social media analytics are accurately tracked and displayed on the online learning platform

Expected Result: Analytics track the reach and engagement of shared content for both learners and instructors on the online learning platform, with accurate and complete data

Severity: Minor

Priority: Low

Type of Testing: Functional Testing



## **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: TC001

Test Case Title: Verify that the app records data on distance, duration, pace, and calories burned

Test Case Description: Test the accuracy and functionality of the activity tracking feature by manually engaging in an activity (e.g. walking) while wearing a wearable device. Compare the data collected by the wearable device with that recorded by the app.

Test Suite: Activity Tracking Module

Test Priority: High

Preconditions:

- User has created an account
- User has wearable device connected
- User is on the main screen of the app

Test Data: Wearable device data (distance, duration, pace, and calories burned)

Test Steps:

1. Begin a walking activity using the wearable device
2. After finishing the walking activity, sync the wearable device data with the app
3. Check the activity summary page for recorded data
4. Compare the app data against the wearable device data

Postconditions:

- The app displays the correct distance
- The app displays the correct duration
- The app displays the correct pace
- The app displays the correct calories burned

Expected Result: The app records accurate data on distance, duration, pace, and calories burned, matching the data from the wearable device.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Data-Driven Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that users can view activity summaries

Test Case Description: Test the presentation and accessibility of activity summaries within the app.

Test Suite: Activity Tracking Module

Test Priority: Medium

Preconditions:

- User has created an account
- User has completed an activity
- User is on the main screen of the app

Test Data: No test data needed

Test Steps:

1. Navigate to the activity history page
2. Select a completed activity
3. Verify that the activity summary is displayed correctly with data on distance, duration, pace, and calories burned

Postconditions:

- The app displays the correct activity summary

Expected Result: The app presents activity summaries accurately and accessibly, enabling users to monitor their physical activity.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that data is stored for historical analysis

Test Case Description: Test the app's ability to store activity data for extended periods for historical analysis and trend monitoring.

Test Suite: Activity Tracking Module

Test Priority: Medium

Preconditions:

- User has created an account
- User has completed multiple activities over several days
- User is on the main screen of the app

Test Data: Multiple activities spanning several days

Test Steps:

1. Navigate to the activity history page
2. Verify that the historical activities are listed with their correct data
3. Perform a data filtering or analysis action

Postconditions:

- The app displays the historical activities
- The app allows data filtering and analysis

Expected Result: The app stores activity data for historical analysis, allowing users to monitor their physical activity trends and progress over time.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that activity tracking can be paused and resumed

Test Case Description: Test the app's ability to pause and resume activity tracking during an ongoing activity.

Test Suite: Activity Tracking Module

Test Priority: Low

Preconditions:

- User has created an account
- User is currently tracking an activity
- User is on the activity tracking screen

Test Data: No test data needed

Test Steps:

1. Start tracking an activity (e.g., running)
2. Pause the activity tracking within the app
3. Perform a pause-unrelated action (e.g., check a message)
4. Resume the activity tracking
5. Stop the activity tracking when finished

Postconditions:

- The app allows pausing and resuming activity tracking

Expected Result: The app enables users to pause and resume activity tracking during ongoing activities, ensuring accurate data collection.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that users receive notifications for milestones

Test Case Description: Test the app's ability to send notifications when users reach user-defined or pre-determined milestones (e.g., distance or calorie milestones).

Test Suite: Activity Tracking Module

Test Priority: Low

Preconditions:

- User has created an account
- User has set a distance or calorie milestone
- User is tracking an activity and closes to reaching the milestone

Test Data: Distance or calorie milestone

Test Steps:

1. Start an activity
2. Approach the milestone (distance or calories burned) during the activity
3. Verify if notification for the milestone appears

Postconditions:

- The app sends a notification when users reach a milestone

Expected Result: The app sends notifications to users when they reach milestones during their activities, encouraging them to stay engaged and motivated.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing, Usability Testing

Test Case Approach: Positive

-----

**Data - 77**

**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: Test the accuracy of GPS tracking by comparing it with a known route and measuring discrepancies between the two.

Test Suite: GPS Tracking

Test Priority: High

Preconditions:

- User has enabled location services.
- User is outdoors with clear sky visibility.

Test Data: A pre-defined route with known waypoints.

Test Steps:

1. Start the Fitness Tracker App.
2. Enable GPS tracking.
3. Begin recording a workout.
4. Follow the pre-defined route.
5. End the workout.
6. Compare the recorded route with the pre-defined route.

Postconditions:

- Recorded route information is saved.
- Comparison data is available for analysis.

Expected Result: Discrepancies between the two routes should be within acceptable tolerance levels.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Performance 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: Test the functionality of viewing detailed maps and elevation profiles for a previously recorded workout route.

Test Suite: GPS Tracking

Test Priority: Medium

Preconditions:

- User has previously recorded a workout with GPS tracking enabled.

Test Data: Previously recorded workout data.

Test Steps:

1. Open the Fitness Tracker App.



2. Navigate to the list of workouts.
3. Select the previously recorded workout.
4. Tap on the "Map" option.
5. Verify that a detailed map is displayed.
6. Tap on the "Elevation Profile" option.
7. Verify that a detailed elevation profile is displayed.

Postconditions:

- Map and elevation profile are displayed correctly.

Expected Result: Users can view detailed maps and elevation profiles for their workout routes.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing, Usability Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: FTA-GPS-003

Test Case Title: Verify that performance data includes location-based metrics

Test Case Description: Test the inclusion of location-based metrics like average speed, maximum speed, and total distance traveled in the workout performance data.

Test Suite: GPS Tracking

Test Priority: Medium

Preconditions:

- User has previously recorded a workout with GPS tracking enabled.

Test Data: Previously recorded workout data.

Test Steps:

1. Open the Fitness Tracker App.
2. Navigate to the list of workouts.

3. Select the previously recorded workout.
4. View the performance data.
5. Check if location-based metrics are included (average speed, maximum speed, total distance traveled).

Postconditions:

- Performance data is displayed correctly.

Expected Result: Performance data includes location-based metrics.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: FTA-GPS-004

Test Case Title: Verify that routes can be saved and revisited

Test Case Description: Test the functionality of saving and revisiting workout routes.

Test Suite: GPS Tracking

Test Priority: High

Preconditions:

- User has previously recorded a workout with GPS tracking enabled.

Test Data: Previously recorded workout data.

Test Steps:

1. Open the Fitness Tracker App.
2. Navigate to the list of workouts.
3. Select the previously recorded workout.
4. Tap on the "Save Route" option.
5. Navigate to the list of saved routes.
6. Verify that the new route is saved.

7. Tap on the saved route.
8. Verify that the details for that route are displayed accurately.

Postconditions:

- Route is saved and displayed correctly.

Expected Result: Users can save and revisit workout routes.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Usability Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: FTA-GPS-005

Test Case Title: Verify that notifications are provided for deviations from planned routes

Test Case Description: Test the functionality of receiving notifications for deviations from a planned route while recording a workout.

Test Suite: GPS Tracking

Test Priority: High

Preconditions:

- User has previously saved a route.
- User has enabled notifications for deviations from the saved route.
- User is recording a workout using the saved route for guidance.

Test Data: Previously saved route data.

Test Steps:

1. Open the Fitness Tracker App.
2. Start a new workout using the saved route for guidance.
3. Intentionally deviate from the saved route.
4. Verify that a notification is received, indicating a deviation from the planned route.

5. Correct the deviation and return to the planned route.

Postconditions:

- Notification is received for the deviation.

Expected Result: Users receive notifications for deviations from their planned routes while recording a workout.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Usability Testing

Test Case Approach: Positive, Negative

-----

**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: TC001

Test Case Title: Verify that heart rate data is recorded in real-time

Test Case Description: This test case verifies the app's ability to capture and display real-time heart rate data using built-in sensors or compatible wearable devices.

Test Suite: Heart Rate Monitoring

Test Priority: High

Preconditions:

1. User has granted heart rate monitoring permission.
2. User is stationary or wearing a compatible device during the test.

Test Data: No test data needed

Test Steps:

1. Launch the Fitness Tracker App.
2. Navigate to the "Heart Rate" section.
3. Ensure that the heart rate sensor is enabled.
4. Remain still or wear a compatible device for 30 seconds.

Postconditions:

1. Heart rate data has been updated within the last 30 seconds.

Expected Result: Heart rate data should be updated and displayed in real-time.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Real-Time Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that users can set and monitor target heart rate zones

Test Case Description: This test case checks whether users can set target heart rate zones and visually monitor their progress during activity.

Test Suite: Heart Rate Monitoring & User Interface

Test Priority: High

Preconditions:

1. User has granted heart rate monitoring permission.
2. User has access to the goal-setting or target heart rate feature.

Test Data: Predefined target heart rate values (e.g., 120 bpm for warm-up, 150 bpm for exercise, and 100 bpm for cool-down)

Test Steps:

1. Launch the Fitness Tracker App.
2. Navigate to the "Goal Setting" or "Target Heart Rate" feature.
3. Enter the predefined target heart rate values.
4. Start heart rate monitoring while exercising.
5. Observe the app's visualization of the user's current heart rate in relation to target zones.

Postconditions:

1. User's heart rate should be displayed according to target zones.

Expected Result: The app should display the user's heart rate within the set target heart rate zones and provide visual cues.

Severity: Major

Priority: High

Type of Testing: Functional Testing, User Interface Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that historical heart rate data is stored and viewable

Test Case Description: This test case verifies whether the app stores historical heart rate data and allows users to view and analyze it.

Test Suite: Heart Rate Monitoring & Data Storage

Test Priority: Medium

Preconditions:

1. User has granted heart rate monitoring permission.
2. User has exercised using the app for at least one week.

Test Data: No test data needed

Test Steps:

1. Launch the Fitness Tracker App.
2. Navigate to the "History" or "Past Data" section.
3. Select a date within the past week for review.
4. Observe the app's display of historical heart rate data.

Postconditions:

1. Historical heart rate data for the selected date should be displayed.

Expected Result: The app should display heart rate data for the selected date, including average, maximum, and minimum heart rate measurements.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Data Storage Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that alerts are provided for abnormal heart rate readings

Test Case Description: This test case checks if the app can detect and provide alerts for abnormal heart rate readings.

Test Suite: Heart Rate Monitoring & Notifications

Test Priority: Medium

Preconditions:

1. User has granted heart rate monitoring permission.
2. User has set up notifications.

Test Data: Predefined abnormal heart rate values (e.g., 200 bpm)

Test Steps:

1. Launch the Fitness Tracker App.
2. Ensure abnormal heart rate alerts are enabled.
3. Simulate an abnormally high heart rate using a dummy device or manual input.
4. Observe the app's notifications or alerts.

Postconditions:

1. An alert should be displayed and/or a notification should be received.

Expected Result: The app should detect the abnormally high heart rate and provide an alert or notification.

Severity:

Priority: Medium

Type of Testing: Functional Testing, Notification Testing

Test Case Approach: Negative

Test Case ID: TC005

Test Case Title: Verify data compatibility with various wearable devices

Test Case Description: This test case verifies whether the app can successfully connect and gather data from a range of compatible wearable devices.



Test Suite: Heart Rate Monitoring & Compatibility

Test Priority: Low

Preconditions:

1. Access to various compatible wearable devices.

Test Data: Compatible wearable devices

Test Steps:

1. Launch the Fitness Tracker App.
2. Pair the app with a compatible wearable device.
3. Start heart rate monitoring using the paired device.
4. Observe the app's display of heart rate data from the connected device.
5. Repeat the process for various other compatible devices.

Postconditions:

1. Heart rate data should be displayed for each connected device.

Expected Result: The app should be able to connect to and gather heart rate data from a variety of compatible devices.

Severity: Major

Priority: Low

Type of Testing: Compatibility Testing, Functional 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: This test case verifies the functionality of allowing users to input their daily food intake and corresponding calorie values.

Test Suite: Calorie Tracking - Food Input

Test Priority: High

Preconditions: User is logged in, user is on the "Add Food" or "Log Food" page.

Test Data: A variety of food items with different calorie values.

Test Steps:

1. Login as a registered user.
2. Navigate to the "Add Food" or "Log Food" page.
3. Enter a food item name in the search bar.
4. Select a food item from the search results.
5. Input the quantity consumed.
6. Tap "Add" or "Log" to submit the entry.

Postconditions: The entered food item and calorie value are displayed in the user's food log.

Expected Result: The user can successfully add food items and calorie values to their daily food 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: This test case ensures that the app accurately calculates and displays total calories consumed and burned for a given time period.

Test Suite: Calorie Tracking - Calorie Calculation

Test Priority: High

Preconditions: User has logged food intake and physical activity data.

Test Data: Previous food intake and physical activity data.

Test Steps:

1. Login as a registered user.
2. Navigate to the "Dashboard" or "Summary" page.
3. Verify that the total calories consumed and burned are displayed.
4. Check that the values are calculated correctly based on user data.

Postconditions: The total calories consumed and burned are correctly displayed.

Expected Result: The user can view the total calories consumed and burned for the selected time period.

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: This test case verifies the user's ability to set daily calorie goals and monitor their progress throughout the day.

Test Suite: Calorie Tracking - Goal Setting

Test Priority: Medium

Preconditions: User is logged in and on the "Settings" or "Goals" page.

Test Data: A daily calorie goal value.

Test Steps:

1. Login as a registered user.
2. Navigate to the "Settings" or "Goals" page.
3. Enter a daily calorie goal value.
4. Save the new calorie goal.
5. Navigate back to the dashboard or summary page.
6. Verify that the daily calorie goal and progress are displayed.

Postconditions: The daily calorie goal is set and progress is displayed.

Expected Result: Users can successfully set and track their daily calorie goals.

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: This test case checks the search functionality for food items and associated calorie information.

Test Suite: Calorie Tracking - Food Database

Test Priority: Medium

Preconditions: User is on the "Add Food" or "Log Food" page.

Test Data: Food items with varying calorie values.

Test Steps:

1. Login as a registered user.
2. Navigate to the "Add Food" or "Log Food" page.
3. Enter a search term for a specific food item.
4. Verify that the food item appears in the search results.
5. Check the associated calorie information for the food item.

Postconditions: The food item is correctly searchable with associated calorie information.

Expected Result: Users can search for food items and find accurate calorie information.

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: This test case ensures that users receive notifications when they exceed their daily calorie goals.

Test Suite: Calorie Tracking - Alert System

Test Priority: Low

Preconditions: User has a daily calorie goal set, and has exceeded it.

Test Data: Previous food intake data.

Test Steps:

1. Login as a registered user.
2. Input food intake data that exceeds the user's daily calorie goal.
3. Verify that an alert is displayed indicating that the user has exceeded their calorie goal.

Postconditions: An alert is displayed when the user exceeds their daily calorie goal.

Expected Result: Users receive notifications when they exceed their daily calorie goals.

Severity: Major

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 Creation Template for Fitness Tracker App - Sleep Tracking Feature:

Test Case ID: TC-ST-01

Test Case Title: Verify that sleep data, including duration and quality, is recorded.

Test Case Description: This test case aims to confirm that the app accurately records sleep data, including sleep duration and quality.

Test Suite: Sleep Tracking

Test Priority: High

Preconditions:

1. User has set up a user account and logged into the app.
2. User has enabled sleep tracking in the app settings.
3. User has worn a wearable device or connected fitness equipment that supports sleep tracking.

Test Data: No test data needed

Test Steps:

1. Use the wearable device or fitness equipment during sleep hours.
2. Synchronize the device or equipment with the Fitness Tracker App.
3. Navigate to the sleep tracking section of the app.
4. Verify sleep data, including duration and quality, is displayed.

Postconditions: Sleep data is accurately recorded and displayed in the app.

Expected Result: Sleep duration and quality are recorded, displayed, and stored correctly.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Integration Testing, System Testing

Test Case Approach: Positive

Test Case ID: TC-ST-02

Test Case Title: Verify that users can view detailed sleep stages and patterns.

Test Case Description: This test case verifies that the app provides detailed insights into sleep stages, such as light sleep, deep sleep, and REM sleep.

Test Suite: Sleep Tracking

Test Priority: High

Preconditions:

1. User has a valid user account and is logged into the app.
2. Sleep tracking is enabled and functioning correctly (as confirmed by TC-ST-01).

Test Data: No test data needed

Test Steps:

1. Navigate to the sleep tracking section of the app.
2. Select a sleep session from the list of recorded sleep data.
3. Verify sleep stages, including light sleep, deep sleep, and REM sleep, are displayed.

Postconditions: Sleep stages are accurately recorded and displayed in the app.

Expected Result: Sleep stages are correctly recorded and displayed in the app.



Severity: Major

Priority: High

Type of Testing: Functional Testing, System Testing

Test Case Approach: Positive

Test Case ID: TC-ST-03

Test Case Title: Verify that the app provides insights into sleep disturbances.

Test Case Description: This test case aims to validate that the app identifies sleep disturbances, such as awakenings during the night or irregular sleep patterns.

Test Suite: Sleep Tracking

Test Priority: Medium

Preconditions:

1. User has a valid user account and is logged into the app.
2. Sleep tracking is enabled and functioning correctly (as confirmed by TC-ST-01).

Test Data: No test data needed

Test Steps:

1. Navigate to the sleep tracking section of the app.
2. Select a sleep session from the list of recorded sleep data.
3. Verify sleep disturbances, such as awakenings and irregular sleep patterns, are displayed.

Postconditions: Sleep disturbances are accurately recorded and displayed in the app.

Expected Result: Sleep disturbances are correctly identified and displayed in the app.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, System Testing

Test Case Approach: Positive

Test Case ID: TC-ST-04

Test Case Title: Verify that users receive recommendations for improving sleep quality.

Test Case Description: This test case verifies that the app provides personalized recommendations based on sleep data, sleep patterns, and sleep disturbances.

Test Suite: Sleep Tracking

Test Priority: Medium

Preconditions:

1. User has a valid user account and is logged into the app.
2. Sleep tracking is enabled and functioning correctly (as confirmed by TC-ST-01).

Test Data: No test data needed

Test Steps:

1. Navigate to the sleep tracking section of the app.
2. Select a sleep session from the list of recorded sleep data.
3. Verify that personalized recommendations for improving sleep quality are provided based on sleep patterns and disturbances.

Postconditions: Personalized recommendations are accurately provided based on sleep data and patterns.

Expected Result: Recommendations for improving sleep quality are personalized, relevant, and actionable.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, System Testing

Test Case Approach: Positive

Test Case ID: TC-ST-05

Test Case Title: Verify that sleep data can be viewed in daily, weekly, and monthly summaries.

Test Case Description: This test case confirms that sleep data, including duration, quality, stages, and disturbances, can be viewed in various timeframes.

Test Suite: Sleep Tracking

Test Priority: Medium

Preconditions:

1. User has a valid user account and is logged into the app.
2. Sleep tracking is enabled and functioning correctly (as confirmed by TC-ST-01).

Test Data: No test data needed

Test Steps:

1. Navigate to the sleep tracking section of the app.
2. Select the timeframe summary option (daily, weekly, or monthly) from the list of recorded sleep data.
3. Verify that sleep data is displayed for the selected timeframe summary.

Postconditions: Sleep data is accurately displayed for the selected timeframe summary.

Expected Result: Sleep data is correctly displayed for daily, weekly, and monthly summaries.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, System 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: FTA-WL001

Test Case Title: Verify that users can log various workout types with details

Test Case Description: This test case verifies that users are able to log their workout activities by specifying exercise type, duration, intensity, and any additional notes or comments.

Test Suite: Workout Logging

Test Priority: High

Preconditions:

- User is logged in
- User has access to workout logging feature

Test Data: Sample workout data (Exercise type: Running, Duration: 30 minutes, Intensity: High)

Test Steps:

1. Navigate to the workout logging feature
2. Select the exercise type as 'Running'
3. Input the duration as '30 minutes'

4. Choose the intensity level as 'High'
5. Add any additional notes if required
6. Save the workout entry

Postconditions:

- Workout entry is saved and displayed in the workout log

Expected Result: Workout entry is saved with the correct exercise type, duration, intensity, and notes.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: FTA-WL002

Test Case Title: Verify that workout logs include duration, intensity, and notes

Test Case Description: This test case aims to confirm that each workout log entry contains the specified duration, intensity, and notes.

Test Suite: Workout Logging

Test Priority: High

Preconditions:

- User is logged in
- User has previously logged a workout

Test Data: Previously logged workout data

Test Steps:

1. Navigate to the workout log
2. Select a previously logged workout entry
3. Verify the duration, intensity, and notes for the selected workout entry

Postconditions:

- Workout log entry details are displayed correctly

Expected Result: Each workout log entry 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: FTA-WL003

Test Case Title: Verify that users can edit and delete workout entries

Test Case Description: This test case verifies that users have the ability to modify or remove their workout log entries.

Test Suite: Workout Logging

Test Priority: High

Preconditions:

- User is logged in
- User has previously logged a workout

Test Data: Previously logged workout data

Test Steps:

1. Navigate to the workout log
2. Select a previously logged workout entry
3. Edit the workout entry (change duration, intensity, or notes)
4. Save the edited workout entry
5. Delete the edited workout entry

Postconditions:

- Workout entry is updated or removed from the workout log

Expected Result: Users can successfully edit and delete their workout log entries.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: FTA-WL004

Test Case Title: Verify that data is stored and viewable for tracking progress

Test Case Description: This test case aims to ensure that workout data is stored properly and can be accessed for tracking progress over time.

Test Suite: Workout Logging

Test Priority: High

Preconditions:

- User is logged in
- User has previously logged multiple workouts

Test Data: Previously logged workout data

Test Steps:

1. Navigate to the workout log
2. Verify that all previously logged workouts are displayed
3. Filter the log by date and workout type
4. Verify that the filtered results match the expected data

Postconditions:

- Workout data is displayed and filterable according to date and workout type

Expected Result: Workout data is stored correctly and can be accessed for progress tracking.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: FTA-WL005

Test Case Title: Verify that the system handles invalid workout data inputs

Test Case Description: This test case verifies that the system handles invalid workout data inputs (e.g., negative duration, invalid exercise type) gracefully.

Test Suite: Workout Logging

Test Priority: Medium

Preconditions:

- User is logged in

Test Data: Invalid workout data (Negative duration, invalid exercise type)

Test Steps:

1. Navigate to the workout logging feature
2. Input invalid workout data (negative duration, invalid exercise type)
3. Attempt to save the workout entry

Postconditions:

- System displays an appropriate error message

Expected Result: The system detects and handles invalid workout data inputs with appropriate error messages.

Severity: Minor

Priority: Medium

Type of Testing: Functional 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 -**

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 1:

Test Case ID: FTA-GS-001

Test Case Title: Verify that users can set goals for various fitness metrics

Test Case Description: Validate the ability for users to create and save personalized fitness goals based on different metrics such as steps, calories, distance, and heart rate.

Test Suite: Goal Setting Feature

Test Priority: High

Preconditions:

- User is logged in.
- User has access to the goal setting feature.

Test Data: User data, fitness metric values

Test Steps:

1. Navigate to the goal setting feature.
2. Select a fitness metric from the list (e.g. steps).
3. Enter a value for the selected fitness metric (e.g. 10000 steps).
4. Save the entered goal.
5. Repeat steps 2-4 for other available fitness metrics.

Postconditions:

- Goals are saved and displayed in the user's goal list.

Expected Result: User can set and save goals for different fitness metrics.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: FTA-GS-002

Test Case Title: Verify that progress tracking is available for each goal

Test Case Description: Check that the app shows real-time progress updates for each user fitness goal.

Test Suite: Progress Tracking Module

Test Priority: High

Preconditions:

- User is logged in.
- User has set fitness goals.

Test Data: User data, fitness metric values, real-time user activity data

Test Steps:

1. Navigate to the goal setting feature.
2. Select a fitness goal from the user's goal list.
3. Perform activities related to the selected fitness goal (e.g. walking for the steps goal).
4. Observe the progress bar or chart as it updates in real-time.

Postconditions:

- Progress is accurately displayed for each fitness goal.

Expected Result: Real-time progress tracking is available for each user fitness goal.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: FTA-GS-003

Test Case Title: Verify that users receive motivational feedback and reminders

Test Case Description: Confirm that the app sends personalized motivational feedback and reminders based on user activity and progress towards fitness goals.

Test Suite: Motivational Feedback Feature

Test Priority: Medium

Preconditions:

- User is logged in.

- User has set fitness goals.
- User has enabled motivational feedback and reminders.

Test Data: User data, fitness metric values, user activity data, predefined motivational messages

Test Steps:

1. Navigate to user profile settings.
2. Enable motivational feedback and reminders.
3. Perform activities related to the user's fitness goals (e.g. walking for the steps goal).
4. Observe motivational messages and reminders as they appear in the app.

Postconditions:

- Motivational messages and reminders are displayed based on user activity and goal progress.

Expected Result: Users receive motivational feedback and reminders based on their fitness goals and activity.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: FTA-GS-004

Test Case Title: Verify that goals can be adjusted or updated as needed

Test Case Description: Ensure that users can edit and update their fitness goals, including changing metric values and goal categories.

Test Suite: Goal Setting Feature

Test Priority: Medium

Preconditions:

- User is logged in.
- User has already set fitness goals.

Test Data: User data, updated fitness metric values

Test Steps:

1. Navigate to the goal setting feature.
2. Select a fitness goal from the user's goal list.
3. Edit the goal by updating the fitness metric value or changing the goal category.
4. Save the edited goal.

Postconditions:

- The edited goal is saved and displayed in the user's goal list.

Expected Result: Users can adjust or update their fitness goals as needed.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: FTA-GS-005

Test Case Title: Verify that the app provides a summary of goal attainment

Test Case Description: Validate that the app shows a summary of the user's completed fitness goals, including metrics achieved and overall progress statistics.

Test Suite: Goal Summary Module

Test Priority: Low

Preconditions:

- User is logged in.
- User has completed fitness goals.

Test Data: User data, completed fitness goals

Test Steps:

1. Navigate to the goal summary feature.
2. Observe the summary of completed fitness goals.
3. Verify that the summary includes achieved fitness metric values, goal categories, and overall progress statistics.

Postconditions:

- Goal summary data is accurately displayed.

Expected Result: The app provides a clear summary of the user's completed fitness goals.

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 ID: TC001

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 pre-designed workout plans based on their fitness levels.

Test Suite: Workout Plans Selection

Test Priority: High

Preconditions:

- User is logged in
- User has access to the workout plans feature

Test Data: User with different fitness level profiles (beginner, intermediate, advanced)

Test Steps:

1. Log in to the app
2. Navigate to the workout plans feature
3. View the list of available workout plans
4. Select a workout plan based on the specified fitness level

Postconditions:

- Selected workout plan is displayed on the user's dashboard

- User can view plan details and start tracking progress

Expected Result:

- User can select a workout plan that matches their fitness level
- The selected plan displays relevant exercise routines and schedules based on the user's fitness level

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that plans include detailed exercise routines and schedules.

Test Case Description: This test case confirms that the selected workout plans include detailed exercise routines and schedules.

Test Suite: Workout Plans Details

Test Priority: High

Preconditions:

- User is logged in
- User has a selected workout plan

Test Data: Selected workout plan with various exercise types

Test Steps:

1. Log in to the app
2. Navigate to the user's dashboard
3. Select the active workout plan
4. Review the exercise routines and schedules for each day of the plan
5. Check the details for at least three random exercises within the plan

Postconditions:

- Exercise routines and schedules are displayed appropriately



- Exercise details are visible and include instructions, images, or videos if applicable

Expected Result:

- Workout plans include detailed exercise routines and schedules
- Exercise details are accurate, complete, and helpful for users executing the workouts

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users can track adherence and progress within the plan.

Test Case Description: This test case verifies that users can effectively track their adherence and progress within the selected workout plan.

Test Suite: Progress Tracking

Test Priority: Medium

Preconditions:

- User is logged in
- User has an active workout plan with at least one completed workout

Test Data: Completed workout data with various exercise types

Test Steps:

1. Log in to the app
2. Navigate to the user's dashboard
3. Select the active workout plan
4. Check the progress tracking feature (e.g., calendar, graph, or chart)
5. Compare the completed workout data with the plan's expected progress

Postconditions:

- Completed workouts and exercises are marked accordingly in the progress tracking system

- Progress is calculated and displayed correctly

Expected Result:

- Users can accurately track their adherence and progress within the workout plan
- Progress tracking feature demonstrates a clear understanding of completed vs. planned workouts

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 notifications and reminders for workouts.

Test Case Description: This test case confirms that the app sends timely notifications and reminders for planned workouts.

Test Suite: Notifications and Reminders

Test Priority: Medium

Preconditions:

- User is logged in
- User has an active workout plan with at least one upcoming workout

Test Data: Planned workout with a specific time and date

Test Steps:

1. Log in to the app
2. Ensure that the notification settings are enabled (if applicable)
3. Leave the app and wait for the scheduled notification time
4. Check the device's notification system for a workout reminder

Postconditions:

- A notification or reminder is successfully triggered for the scheduled workout

Expected Result:

- Users receive timely notifications and reminders for their planned workouts

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

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 workout plans according to their progress and personal preferences.

Test Suite: Plan Adjustment

Test Priority: Low

Preconditions:

- User is logged in
- User has an active workout plan

Test Data: Completed workout plan with various progress levels, different exercise preferences

Test Steps:

1. Log in to the app
2. Navigate to the user's dashboard
3. Select the active workout plan
4. Modify the plan by adjusting the exercise routines, schedules, or fitness goals
5. Save the changes and continue using the adjusted plan

Postconditions:

- Modified workout plan is saved and displayed on the user's dashboard

Expected Result:

- Users can successfully adjust workout plans based on their progress and preferences

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

-----

## **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 that a registered user can search for, request to join, and participate in community groups within the Fitness Tracker App.

Test Suite: Community Module

Test Priority: High

Preconditions:

- User is logged into the Fitness Tracker App
- User has access to the community feature

Test Data: User account with login credentials, community group name

Test Steps:

1. Log into the Fitness Tracker App
2. Navigate to the community feature
3. Search for a community group using the group name
4. Request to join the community group
5. Wait for approval from the group administrator
6. Access the community group once approved
7. Post a message or comment within the community group

Postconditions:

- User has joined and participated in a community group

Expected Result: User can successfully search for, join, and participate in community groups within the Fitness Tracker App.

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 that a user can share their progress updates and achievements with other users in their community groups.

Test Suite: Community Module

Test Priority: Medium

Preconditions:

- User is logged into the Fitness Tracker App
- User has access to the community feature
- User has joined at least one community group

Test Data: User account with login credentials, progress update or achievement data

Test Steps:

1. Log into the Fitness Tracker App
2. Navigate to the community feature
3. Access a community group the user has joined
4. Select the option to share progress or achievements
5. Enter the progress update or achievement details
6. Post the progress update or achievement to the community group

Postconditions:

- Progress update or achievement is visible to other community group members

Expected Result: User can successfully share progress updates and achievements with other users in their community groups.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC003

Test Case Title: Verify that the app facilitates participation in community challenges

Test Case Description: This test case verifies that the Fitness Tracker App fosters participation in community challenges among community group members.

Test Suite: Community Module

Test Priority: Medium

Preconditions:

- User is logged into the Fitness Tracker App
- User has access to the community feature
- User has joined at least one community group

Test Data: User account with login credentials, community challenge data

Test Steps:

1. Log into the Fitness Tracker App
2. Navigate to the community feature
3. Access a community group the user has joined
4. Choose to participate in a community challenge
5. Follow the prompts to complete the community challenge
6. Submit the completed community challenge for verification

Postconditions:

- Community challenge participation is recorded and visible in the community group

Expected Result: User can successfully participate in community challenges and have their participation recorded within 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 users can offer and receive support and encouragement

Test Case Description: This test case verifies that users can offer and receive support and encouragement within community groups.

Test Suite: Community Module

Test Priority: Low

Preconditions:

- User is logged into the Fitness Tracker App
- User has access to the community feature
- User has joined at least one community group

Test Data: User account with login credentials, support or encouragement message

Test Steps:

1. Log into the Fitness Tracker App
2. Navigate to the community feature
3. Access a community group the user has joined
4. Identify a post or comment from another user
5. Select the option to offer support or encouragement
6. Enter a supportive or encouraging message
7. Post the message within the community group

Postconditions:

- Supportive or encouraging message is visible to other community group members



Expected Result: User can successfully offer and receive support and encouragement within community groups.

Severity: Major

Priority: Low

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: This test case verifies that community activity and engagement are tracked within the Fitness Tracker App.

Test Suite: Community Module

Test Priority: Low

Preconditions:

- User is logged into the Fitness Tracker App
- User has access to the community feature
- User has joined at least one community group

Test Data: User account with login credentials, community activity data

Test Steps:

1. Log into the Fitness Tracker App
2. Navigate to the community feature
3. Access a community group the user has joined
4. Engage in community activity (e.g., join a challenge, post a message)
5. Log out and wait for a predetermined amount of time (e.g., 24 hours)
6. Log back into the Fitness Tracker App

Postconditions:

- Community activity and engagement are visible within the user's community group

Expected Result: Community activity and engagement are tracked within the Fitness Tracker App.

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

-----

**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 ID: TC001

Test Case Title: Verify that the app supports integration with multiple wearable devices

Test Case Description: This test case verifies that the Fitness Tracker App can connect and sync data from various wearable devices, ensuring compatibility and support for multiple brands and models.

Test Suite: Wearable Device Integration

Test Priority: High

Preconditions:

- Wearable devices are properly connected to the internet and in range of the mobile device.
- The Fitness Tracker App is installed on a mobile device.

Test Data: Fitbit Charge 4, Garmin Venu, Apple Watch Series 6, Samsung Gear Fit2 Pro, and Huawei Watch GT2

Test Steps:

1. Open the Fitness Tracker App on the mobile device.
2. Navigate to the device integration settings.
3. Enable integration for each wearable device.
4. Confirm successful connection for each device.
5. Sync data from each device.

Postconditions:

- Data from all connected devices should be visible in the Fitness Tracker App.

Expected Result: The Fitness Tracker App should successfully connect, sync data, and display information from all specified wearable devices.

Severity: Major

Priority: High

Type of Testing: Integration Testing, Compatibility Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that users can sync activity data from their wearables

Test Case Description: This test case checks that users can manually sync and view activity data from their connected wearable devices within the Fitness Tracker App.

Test Suite: Wearable Device Integration

Test Priority: High

Preconditions:

- Wearable devices are properly connected to the internet and in range of the mobile device.
- The Fitness Tracker App is installed on a mobile device.
- A wearable device is connected and displayed in the app.

Test Data: Previously recorded activity data on the wearable device

Test Steps:

1. Open the Fitness Tracker App on the mobile device.
2. Navigate to the device integration section.
3. Select a connected wearable device.
4. Initiate a manual sync of the wearable device data.
5. Verify that the synced data is displayed in the Fitness Tracker App.

Postconditions:

- Sync confirmation message is displayed.
- Synced data is visible and accurate in the Fitness Tracker App.

Expected Result: Users can manually sync and view activity data from their connected wearable devices within the Fitness Tracker App.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Integration Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that the app provides real-time data updates from wearables

Test Case Description: This test case ensures that the Fitness Tracker App receives and displays real-time data updates from connected wearable devices during usage.

Test Suite: Real-time Data Updates

Test Priority: High

Preconditions:

- Wearable devices are properly connected to the internet and in range of the mobile device.
- The Fitness Tracker App is installed on a mobile device.
- A wearable device is connected and displayed in the app.

Test Data: No test data needed

Test Steps:

1. Open the Fitness Tracker App on the mobile device.
2. Start tracking a workout or activity on the connected wearable device.
3. Monitor the Fitness Tracker App for real-time updates of the activity data (e.g., steps, heart rate, and calories burned).
4. Stop the workout or activity on the connected wearable device.
5. Verify that the final activity data is displayed in the Fitness Tracker App.

Postconditions:

- Real-time data updates are displayed during the activity.
- Final activity data is accurate and visible in the Fitness Tracker App.

Expected Result: The Fitness Tracker App provides real-time data updates from connected wearable devices during usage.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Integration Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that wearable data includes steps, heart rate, and calories burned

Test Case Description: This test case confirms that the Fitness Tracker App displays the required wearable data, including steps taken, heart rate, and calories burned, after syncing.

Test Suite: Wearable Data Display

Test Priority: Medium

Preconditions:

- Wearable devices are properly connected to the internet and in range of the mobile device.
- The Fitness Tracker App is installed on a mobile device.
- A wearable device is connected and displayed in the app.
- Activity data has been synced from the wearable device.

Test Data: Previously recorded activity data on the wearable device

Test Steps:

1. Open the Fitness Tracker App on the mobile device.
2. Navigate to the activity data section.
3. Verify that steps, heart rate, and calories burned data points are visible.
4. Compare the data points with the data from the wearable device.

Postconditions:

- Steps, heart rate, and calories burned data points are visible and accurate.

Expected Result: The Fitness Tracker App displays steps, heart rate, and calories burned data points after syncing with the wearable device.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that users can troubleshoot connectivity issues with wearables

Test Case Description: This test case verifies that users can identify and resolve connectivity issues between their mobile device and wearable devices within the Fitness Tracker App.

Test Suite: Connectivity Troubleshooting

Test Priority: Medium

Preconditions:

- Wearable devices are in close proximity to the mobile device but not connected.
- The Fitness Tracker App is installed on a mobile device.

Test Data: No test data needed

Test Steps:

1. Open the Fitness Tracker App on the mobile device.
2. Navigate to the device integration settings.
3. Try to connect a wearable device that is not currently connected.
4. Observe the connection status and any error messages.
5. Follow the troubleshooting steps provided by the app (e.g., checking internet connection, restarting devices, etc.).
6. Attempt to reconnect the wearable device.

Postconditions:

- Connectivity issues should be resolved or clear instructions for resolution should be provided.

Expected Result: Users can troubleshoot connectivity issues with wearable devices within the Fitness Tracker App and either resolve the issues or receive clear guidance for resolution.

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing, User Interface Testing

Test Case Approach: Negative

-----

**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\_WaterIntakeTracking

Test Case Title: Verify that users can log their daily water intake

Test Case Description: This test case verifies that users can successfully enter their water intake data into the Fitness Tracker App.

Test Suite: Water Intake Tracking

Test Priority: High

Preconditions:

- User is registered and logged into the app
- User is on the "Water Intake" page

Test Data: 500ml (or any valid water intake value)

Test Steps:

1. Enter water intake value (e.g. 500ml) in the input field
2. Tap the "Log" or "Save" button
3. Observe the updated water intake total displayed on the screen

Postconditions:

- Water intake logs are saved and displayed in the user's history

Expected Result: The entered water intake value is accurately logged and displayed in the user's history.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002\_WaterIntakeCalculation

Test Case Title: Verify that the app calculates total water consumption for the day

Test Case Description: This test case verifies that the app accurately calculates and displays the user's total water intake for the day after logging multiple entries.

Test Suite: Water Intake Tracking

Test Priority: High

Preconditions:

- User is registered and logged into the app
- User is on the "Water Intake" page
- User has logged at least two water intake entries

Test Data: 500ml, 300ml

Test Steps:

1. Enter water intake value (e.g. 500ml) in the input field
2. Tap the "Log" or "Save" button
3. Repeat steps 1 and 2 with different water intake value (e.g. 300ml)
4. Observe the updated total water intake displayed on the screen

Postconditions:

- Total water intake for the day is accurately calculated

Expected Result: Total water intake for the day (500ml + 300ml) is accurately calculated and displayed in the user's history.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003\_HydrationGoalSetting

Test Case Title: Verify that users can set daily hydration goals

Test Case Description: This test case verifies that users can successfully set and modify their daily hydration goals in the Fitness Tracker App.

Test Suite: Water Intake Tracking

Test Priority: Medium

Preconditions:

- User is registered and logged into the app
- User is on the "Water Intake" page

Test Data: 2000ml

Test Steps:

1. Access hydration goal settings from the Water Intake page
2. Enter desired hydration goal value (e.g. 2000ml) in the input field
3. Tap "Save" or "Confirm" to save the new hydration goal
4. Repeat steps 2 and 3 with a new value to modify the hydration goal

Postconditions:

- User's daily hydration goals are saved and displayed in the app

Expected Result: User's daily hydration goals are successfully saved and displayed; they can be modified as needed.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004\_WaterIntakeReminders

Test Case Title: Verify that reminders are provided to encourage regular water intake

Test Case Description: This test case verifies that the Fitness Tracker App provides reminders at configured intervals to encourage users to drink water.

Test Suite: Water Intake Tracking

Test Priority: Low

Preconditions:

- User is registered and logged into the app
- User has enabled water intake reminders
- User has set a daily hydration goal

#### Test Steps:

1. Configure reminder interval (e.g. every 2 hours)
2. Wait for the configured reminder interval to elapse
3. Observe if a reminder is displayed or received (e.g. push notification)

#### Postconditions:

- Reminder is sent at the configured interval

Expected Result: Reminders are sent at the configured interval, encouraging the user to drink water.

Severity: Major

Priority: Low

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005\_HydrationProgressViewing

Test Case Title: Verify that users can view hydration progress and trends

Test Case Description: This test case verifies that users can view their hydration progress and historical trends in the Fitness Tracker App.

Test Suite: Water Intake Tracking

Test Priority: Medium

#### Preconditions:

- User is registered and logged into the app
- User has logged at least two water intake entries

#### Test Steps:

1. Access hydration progress from the Water Intake page
2. Observe if a graph, chart, or table displaying hydration progress and trends is available
3. Verify that data points include date, water intake, and progress towards hydration goal

#### Postconditions:

- Hydration progress is clearly and accurately displayed

Expected Result: Users can view their hydration progress and trends with relevant data points, allowing them to monitor their intake over time.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Note: While designing the test cases, I've ensured contextual relevance, comprehensive coverage, structured response, and quality assurance focus, incorporating the project context and user story details. Each acceptance criterion is covered by at least one test case, with edge cases included where applicable. Additionally, test case priorities are determined based on risk and impact to the project.

-----

**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: FTN-001

Test Case Title: Verify that users can log their daily meals and snacks.

Test Case Description: Validate the ability for users to add meals and snacks to their food diary within the Fitness Tracker App.

Test Suite: Nutrition Tracking - Food Diary

Test Priority: High

Preconditions:

- User has a registered account.
- User is logged into the Fitness Tracker App.

Test Data: Breakfast, lunch, and dinner items consisting of a mix of proteins, carbohydrates, and fats.

Test Steps:

1. Log into the Fitness Tracker App.
2. Navigate to the food diary section.
3. Select the option to log a meal or snack.
4. Enter the meal name, time, and food items consumed.
5. Save the entered meal or snack.

Postconditions:

- Meal or snack is saved and displayed in the user's food diary.

Expected Result: Meals and snacks are successfully logged and visible in the food diary with accurate time stamps.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: FTN-002

Test Case Title: Verify that the app calculates macronutrient intake (carbs, fats, proteins).

Test Case Description: Validate the Fitness Tracker App's ability to calculate and display macronutrient intake from logged meals and snacks.

Test Suite: Nutrition Tracking - Macronutrient Calculation

Test Priority: High

Preconditions:

- User has a registered account.
- User is logged into the Fitness Tracker App.
- User has logged meals and snacks with nutritional information.

Test Data: Pre-entered meals and snacks with known macronutrient content.

Test Steps:

1. Log into the Fitness Tracker App.
2. Navigate to the food diary section.
3. Verify that total carbohydrates, fats, and proteins are displayed for each meal and snack.
4. Check that the app calculates and displays the daily macronutrient intake totals.

Postconditions:

- Macronutrient values are accurately calculated and displayed for both individual meals/snacks and the entire day.

Expected Result: The app calculates and displays accurate macronutrient intake values for both individual

meals/snacks and the entire day.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: FTN-003

Test Case Title: Verify that users can set nutritional goals and track progress.

Test Case Description: Validate the ability for users to set nutritional goals and track their progress towards these goals within the Fitness Tracker App.

Test Suite: Nutrition Tracking - Goal Setting and Tracking

Test Priority: Medium

Preconditions:

- User has a registered account.
- User is logged into the Fitness Tracker App.

Test Data: Pre-defined nutritional goals (e.g., daily caloric intake, macronutrient ratios).

Test Steps:

1. Log into the Fitness Tracker App.
2. Navigate to the goals section.
3. Set daily caloric intake and macronutrient ratio goals.
4. Save the set goals.
5. Log meals and snacks throughout the day.
6. Navigate to the progress tracking section.
7. Verify that the app calculates and displays progress towards the set nutritional goals.

Postconditions:

- Nutritional goal progress is accurately calculated and displayed to the user.

Expected Result: The app calculates and displays accurate progress towards the user's set nutritional goals.



Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: FTN-004

Test Case Title: Verify that the app provides nutritional information and suggestions.

Test Case Description: Validate the ability for the Fitness Tracker App to provide relevant and accurate nutritional information based on logged meals and snacks.

Test Suite: Nutrition Tracking - Nutritional Information and Suggestions

Test Priority: Medium

Preconditions:

- User has a registered account.
- User is logged into the Fitness Tracker App.
- User has logged meals and snacks with nutritional information.

Test Data: Pre-entered meals and snacks with varied nutritional content.

Test Steps:

1. Log into the Fitness Tracker App.
2. Navigate to the food diary section.
3. Verify that the app displays relevant nutritional information (e.g., total caloric intake, macronutrient percentages) based on logged meals and snacks.
4. Check for personalized suggestions based on the user's nutritional intake (e.g., recommending different meal options for a balanced diet).

Postconditions:

- Accurate nutritional information is displayed based on the user's logged meals and snacks.
- Relevant and personalized suggestions are provided for a more balanced diet.

Expected Result: The app provides accurate nutritional information and personalized suggestions for a more

balanced diet based on the user's logged meals and snacks.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: FTN-005

Test Case Title: Verify that users can view their nutritional history and trends.

Test Case Description: Validate that users have access to and can review their nutritional history and trends within the Fitness Tracker App.

Test Suite: Nutrition Tracking - Nutritional History and Trends

Test Priority: Low

Preconditions:

- User has a registered account.
- User is logged into the Fitness Tracker App.
- User has logged meals and snacks for at least one week.

Test Steps:

1. Log into the Fitness Tracker App.
2. Navigate to the nutritional history section.
3. Verify that the app displays a historical record of the user's logged meals and snacks with accurate nutritional information (total daily caloric intake, macronutrient percentages, etc.).
4. Check for visual representation of nutritional trends (e.g., graphs, charts) over time.

Postconditions:

- Nutritional history and trends are accurately displayed over the selected time period.

Expected Result: The Fitness Tracker App displays accurate nutritional history and visual representations of nutritional trends over time.

Severity: Minor

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 ID: TC001

Test Case Title: Verify that users receive weekly and monthly progress reports

Test Case Description: This test case verifies that the system generates and delivers weekly and monthly progress reports to the user.

Test Suite: Progress Reports

Test Priority: High

Preconditions:

- User has an active account
- User has logged workout data

Test Data: User data with workout activities

Test Steps:

1. Log in to the Fitness Tracker App
2. Access user profile settings
3. Set preferred report frequency to weekly and monthly
4. Perform workout activities for a week and a month
5. Wait for the report generation schedule

Postconditions:

- Weekly and monthly progress reports are delivered to the user's inbox or notification center

Expected Result:

- The system generates and sends weekly and monthly progress reports containing user workout data.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that reports include activity levels, workout performance, and goal attainment

Test Case Description: This test case ensures that the generated progress reports contain accurate and complete information on user activity levels, workout performance, and goal attainment.

Test Suite: Progress Reports

Test Priority: High

Preconditions:

- User has an active account
- User has logged workout data with activity levels, workout performance, and goal attainment

Test Data: User data with workout activities, goals, and achievements

Test Steps:

1. Log in to the Fitness Tracker App
2. Access user profile settings
3. Set preferred report frequency to weekly or monthly
4. Perform workout activities for a week or a month
5. Verify activity levels, workout performance, and goal attainment in generated reports

Postconditions:

- Reports accurately display user activity levels, workout performance, and goal attainment

Expected Result:

- The system generates reports with accurate and complete information regarding user activity levels, workout performance, and goal attainment.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users can view detailed breakdowns of achievements and trends

Test Case Description: This test case checks if users can access a detailed breakdown of their achievements and trends within generated progress reports.

Test Suite: Progress Reports

Test Priority: Medium

Preconditions:

- User has an active account
- User has logged workout data with achievements and trends

Test Data: User data with workout activities, achievements, and trends

Test Steps:

1. Log in to the Fitness Tracker App
2. Access user profile settings
3. Set preferred report frequency to weekly or monthly
4. Perform workout activities with achievements and trends for a week or a month
5. Locate and open the progress report
6. Navigate to the achievements and trends section

Postconditions:

- Users can view detailed breakdowns of their achievements and trends

Expected Result:

- The system allows users to view a detailed breakdown of their achievements and trends within generated progress reports.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that the app provides recommendations based on report findings

Test Case Description: This test case verifies that the Fitness Tracker App gives personalized recommendations based on the user's progress report findings.

Test Suite: Progress Reports

Test Priority: Medium

Preconditions:

- User has an active account
- User has logged workout data
- User has a progress report with findings

Test Data: User data with workout activities, goals, and a generated progress report

Test Steps:

1. Log in to the Fitness Tracker App
2. Access user profile settings
3. Set preferred report frequency to weekly or monthly
4. Perform workout activities for a week or a month
5. Generate and open the progress report
6. Scroll to the recommendations section

Postconditions:

- Recommendations are displayed based on user progress report findings

Expected Result:

- The system provides personalized recommendations for users based on their progress report findings.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Regression Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that users can customize the frequency and content of reports

Test Case Description: This test case checks if users can customize the frequency and content of their progress reports.

Test Suite: Progress Reports

Test Priority: Low

Preconditions:

- User has an active account

Test Data: User data with workout activities

Test Steps:

1. Log in to the Fitness Tracker App
2. Access user profile settings
3. Modify the preferred report frequency and content
4. Save the changes
5. Perform workout activities
6. Generate a progress report
7. Verify that the generation frequency and content match the user's customizations

Postconditions:

- The report frequency and content reflect user customization

Expected Result:

- The system allows users to customize the frequency and content of their progress reports.

Severity: Minor

Priority: Low

Type of Testing: Functional Testing, Regression 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 -**

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: As a user, I want to be able to participate in different fitness challenges within the app. This test case verifies that the app allows users to join multiple in-app challenges.

Test Suite: In-App Challenges

Test Priority: High

Preconditions:

- User is logged in
- User is on the challenges page

Test Data: No test data needed

Test Steps:

1. Tap on the "Challenges" tab
2. Scroll through the list of available challenges
3. Select any challenge and tap "Join"
4. Confirm that the challenge has been added to the user's active challenges list

Postconditions:

- Selected challenge is added to the user's active challenges list
- User receives a confirmation message that they have joined the challenge

Expected Result:

- Users can join multiple in-app challenges without encountering any issues

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: As a user, I want to understand the goals and rules of each challenge. This test case verifies that each challenge has clear and concise objectives and rules.

Test Suite: In-App Challenges

Test Priority: High

Preconditions:

- User is logged in
- User is on the challenges page

Test Data: No test data needed

Test Steps:

1. Tap on the "Challenges" tab
2. Select any challenge
3. View the challenge details, including objectives and rules
4. Verify that objectives and rules are clear and easy to understand

Postconditions: N/A

Expected Result:

- Each challenge has clear objectives and rules, enabling users to participate effectively

Severity: Major

Priority: High

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: As a user, I want to monitor my challenge progress. This test case verifies that users can track their progress toward achieving the objectives of each challenge.

Test Suite: In-App Challenges

Test Priority: High

Preconditions:

- User is logged in

- User has joined at least one active challenge

Test Data: No test data needed

Test Steps:

1. Tap on the "Challenges" tab
2. Select an active challenge
3. View the progress meter or tracking information for that challenge
4. Perform an action that contributes to challenge progress (e.g., complete a workout, log a meal)
5. Verify that challenge progress has been updated accordingly

Postconditions: N/A

Expected Result:

- Users can track their progress within a challenge and see updated information after performing relevant actions

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: As a user, I want to earn rewards for completing challenges or reaching milestones.

This test case verifies that users receive rewards upon completing specified challenge milestones.

Test Suite: In-App Challenges

Test Priority: High

Preconditions:

- User is logged in

- User has joined an active challenge with milestones
- User has completed all necessary actions to reach at least one milestone

Test Data: No test data needed

Test Steps:

1. Tap on the "Challenges" tab
2. Select the active challenge with completed milestones
3. Verify that the reward(s) associated with the completed milestone(s) have been awarded

Postconditions: N/A

Expected Result:

- Users receive rewards when they complete specified milestones within a challenge

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: As a user, I want to compete with my friends and see where I stand in the challenge rankings. This test case verifies that users can view leaderboards and compete against their friends within in-app challenges.

Test Suite: In-App Challenges

Test Priority: High

Preconditions:

- User is logged in
- User has joined an active challenge with friends

- Friends have also joined the same active challenge

Test Data: No test data needed

Test Steps:

1. Tap on the "Challenges" tab
2. Select the active challenge in which the user's friends are participating
3. View the leaderboard,
4. Verify that the user's ranking among their friends is visible and accurate

Postconditions: N/A

Expected Result:

- Users can view leaderboards and compete against their friends within in-app challenges

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

-----

**Data - 90**

**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 ID: TC001\_CustomizableDashboards

Test Case Title: Verify that users can select which metrics to display on their dashboard

Test Case Description: This test case ensures that the app allows users to customize the fitness metrics displayed on their dashboard.

Test Suite: Customizable Dashboards

Test Priority: High

Preconditions:

- User is logged into the Fitness Tracker App
- User has access to the dashboard customization settings

Test Data: A list of available fitness metrics (steps, calories burned, distance traveled, heart rate, etc.)

Test Steps:

1. Log into the Fitness Tracker App
2. Navigate to the dashboard customization settings
3. Select a fitness metric from the list of available options

4. Add the selected fitness metric to the dashboard

5. Repeat steps 3-4 for additional metrics

Postconditions:

- Selected fitness metrics are displayed on the user's dashboard

Expected Result: The dashboard displays the user-selected fitness metrics.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC002\_DashboardLayoutCustomization

Test Case Title: Verify that the app allows customization of dashboard layout and theme

Test Case Description: This test case checks if the Fitness Tracker App permits users to modify the layout and theme of their dashboard according to their preferences.

Test Suite: Customizable Dashboards

Test Priority: High

Preconditions:

- User is logged into the Fitness Tracker App
- User has access to the dashboard customization settings

Test Data: Different layout options and themes

Test Steps:

1. Log into the Fitness Tracker App
2. Navigate to the dashboard customization settings
3. Choose a new layout for the dashboard
4. Apply the selected layout
5. Select a new theme for the dashboard
6. Apply the chosen theme



Postconditions:

- Dashboard layout is updated with the selected option
- Dashboard theme is changed to the chosen design

Expected Result: The dashboard displays the newly selected layout and theme.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003\_SaveDashboardSettings

Test Case Title: Verify that users can save and update their dashboard settings

Test Case Description: This test case confirms that the Fitness Tracker App enables users to save their customized dashboard settings and update them as required.

Test Suite: Customizable Dashboards

Test Priority: Medium

Preconditions:

- User is logged into the Fitness Tracker App
- User has access to the dashboard customization settings

Test Data: Previously customized dashboard settings

Test Steps:

1. Log into the Fitness Tracker App
2. Navigate to the dashboard customization settings
3. Modify the existing dashboard settings (add/remove metrics, change layout/theme)
4. Save the updated settings

Postconditions:

- Modified dashboard settings are saved

Expected Result: The dashboard displays the updated settings, which are saved for future use.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC004\_RealTimeDisplay

Test Case Title: Verify that dashboard displays are dynamic and update in real-time

Test Case Description: This test case ensures that the Fitness Tracker App's dashboard dynamically updates in real-time as users input new fitness data.

Test Suite: Customizable Dashboards

Test Priority: Medium

Preconditions:

- User is logged into the Fitness Tracker App
- User has access to the dashboard
- User has input fitness data during the current session

Test Data: None

Test Steps:

1. Log into the Fitness Tracker App
2. Perform an activity that generates fitness data (e.g., walk, run)
3. Observe the dashboard metrics to check if they update in real-time

Postconditions:

- Fitness data is displayed on the dashboard as it is generated

Expected Result: The dashboard dynamically updates to reflect new fitness data in real-time.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Non-Functional Testing

Test Case Approach: Positive

Test Case ID: TC005\_ResetDashboard

Test Case Title: Verify that users can reset the dashboard to default settings

Test Case Description: This test case confirms the Fitness Tracker App allows users to reset their dashboard to the default settings.

Test Suite: Customizable Dashboards

Test Priority: Low

Preconditions:

- User is logged into the Fitness Tracker App
- User has access to the dashboard customization settings

Test Data: None

Test Steps:

1. Log into the Fitness Tracker App
2. Navigate to the dashboard customization settings
3. Select the option to reset the dashboard to default settings
4. Confirm the action to reset the dashboard

Postconditions:

- Dashboard is restored to the default settings

Expected Result: The dashboard displays the default fitness metrics, layout, and theme.

Severity: Major

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 -**

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: TC001

Test Case Title: Verify that users can share workout summaries and achievements

Test Case Description: As a user, I want to share my fitness achievements on social media so that I can inspire others and stay motivated. This test case verifies that the app allows users to share workout summaries and achievements on social media platforms.

Test Suite: Social Sharing

Test Priority: High

Preconditions: User must have a valid social media account linked to the app, user must complete a workout or achieve a milestone.

Test Data: Workout summary or achievement data

Test Steps:

1. Complete a workout or reach a milestone within the app.
2. Navigate to the workout summary or achievements section.
3. Select the sharing option.
4. Choose the desired social media platform.
5. Customize the content and format of the shared post if necessary.
6. Post the update.

Postconditions: Workout summary or achievement is shared on the selected social media platform.

Expected Result: Workout summaries and achievements are shared accurately, displaying relevant data and user-defined content on the chosen social media platform.

Severity: Major

Priority: High

Type of Testing: Functional Testing, Social Media Integration Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that the app supports sharing on multiple social media platforms

Test Case Description: As a user, I want to share my fitness achievements on various social media platforms to reach a wider audience. This test case verifies that the app supports sharing on multiple social media platforms.

Test Suite: Social Sharing

Test Priority: Medium

Preconditions: User must have valid accounts linked to the app for at least three different social media platforms.

Test Data: Account credentials, workout summary, or achievement data.

Test Steps:

1. Complete a workout or reach a milestone within the app.
2. Navigate to the workout summary or achievements section.
3. Select the sharing option.
4. Choose three different social media platforms.
5. Customize the content and format of the shared post if necessary.
6. Post the updates.

Postconditions: Workout summaries or achievements are shared accurately on all three chosen social media platforms.

Expected Result: The app successfully shares workout summaries and achievements on multiple social media platforms, displaying relevant data and user-defined content.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Social Media Integration Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users can customize the content and format of shared posts

Test Case Description: As a user, I want to personalize my shared fitness achievements on social media so that they represent my unique style and personality. This test case verifies that users can customize the content and format of shared posts.

Test Suite: Social Sharing

Test Priority: High

Preconditions: User has completed a workout or reached a milestone, and has customized sharing options

available within the app.

Test Data: Workout summary, achievement data, text, images, or hashtags.

Test Steps:

1. Complete a workout or reach a milestone within the app.
2. Navigate to the workout summary or achievements section.
3. Select the sharing option.
4. Choose a social media platform.
5. Customize the content of the shared post, including text, images, or hashtags.
6. Format the shared post as desired (e.g., bold text, bullet points).
7. Post the update.

Postconditions: Workout summary or achievement is shared on the selected social media platform with customized content and format.

Expected Result: The app allows users to customize the content and format of shared posts, including text, images, hashtags, and formatting, before sharing on social media platforms.

Severity: Major

Priority: High

Type of Testing: Functional Testing, User Interface Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that sharing features are integrated seamlessly with social media platforms

Test Case Description: As a user, I want the sharing process to be smooth, intuitive, and integrated with social media platforms. This test case verifies that sharing features are integrated seamlessly with social media platforms.

Test Suite: Social Sharing

Test Priority: High

Preconditions: User has completed a workout or reached a milestone and has linked at least one social media

account to the app.

Test Data: Workout summary or achievement data.

Test Steps:

1. Complete a workout or reach a milestone within the app.
2. Navigate to the workout summary or achievements section.
3. Select the sharing option.
4. Choose a social media platform.
5. Customize the content of the shared post if desired.
6. Post the update.
7. Verify the shared post within the chosen social media platform.

Postconditions: Workout summary or achievement is shared accurately on the chosen social media platform.

Expected Result: The app's sharing features are seamlessly integrated with social media platforms, allowing users to easily share workout summaries and achievements without encountering technical issues or complications.

Severity: Critical

Priority: High

Type of Testing: Functional Testing, Social Media Integration Testing, Usability Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that users can track the engagement of their shared posts

Test Case Description: As a user, I want to monitor the impact of my shared fitness achievements on social media so that I can gauge their effectiveness at inspiring others and staying motivated. This test case verifies that users can track the engagement of their shared posts.

Test Suite: Social Sharing

Test Priority: Medium

Preconditions: User has shared a workout summary or achievement on at least one social media platform.



Test Data: Shared post data (e.g., likes, comments, shares, views).

Test Steps:

- 1. Share a workout summary or achievement on a social media platform through the app.
- 2. Navigate to the shared post within the social media platform.
- 3. Review and record the engagement metrics (e.g., likes, comments, shares, views).
- 4. Navigate to the app's engagement tracking section.
- 5. Verify that the shared post and its engagement metrics are accurately displayed within the app.

Postconditions: The app accurately displays the engagement metrics for the shared post.

Expected Result: The app allows users to track the engagement of their shared posts, displaying accurate and up-to-date metrics for likes, comments, shares, and views.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, Social Media Integration Testing

Test Case Approach: Positive

-----

**Data - 92**

**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: TRR-001

Test Case Title: Verify that users can log rest days and recovery activities.

Test Case Description: As a user, I want to be able to log rest days and recovery activities to track my rest and recovery.

Test Suite: Rest and Recovery Tracking

Test Priority: High

Preconditions:

1. User is logged in.
2. User has access to the rest and recovery tracking feature.

Test Data: Rest day or recovery activity data (e.g., type, duration, date)

Test Steps:

1. Navigate to the rest and recovery tracking feature.
2. Select the option to log a rest day or recovery activity.

3. Enter the required information for the rest day or recovery activity.

4. Save the rest day or recovery activity log.

Postconditions:

1. The rest day or recovery activity log is saved and accessible in the user's recovery history.

Expected Result: The user can successfully log rest days and recovery activities.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 2:

Test Case ID: TRR-002

Test Case Title: Verify that the app tracks metrics such as resting heart rate and sleep quality.

Test Case Description: As a user, I want the app to automatically track rest and recovery metrics like resting heart rate and sleep quality.

Test Suite: Rest and Recovery Tracking

Test Priority: High

Preconditions:

1. User is logged in.

2. User has a wearable device or fitness equipment integrated with the app.

3. User has worn the wearable device or used the fitness equipment during sleep.

Test Data: Sleep duration, resting heart rate data.

Test Steps:

1. Navigate to the rest and recovery tracking feature.

2. Select the night's sleep or resting heart rate data for which metrics should be tracked.

Postconditions:

1. Sleep quality and resting heart rate metrics are displayed in the user's recovery history.

Expected Result: The app accurately tracks and displays rest and recovery metrics such as sleeping quality and resting heart rate.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 3:

Test Case ID: TRR-003

Test Case Title: Verify that users receive recovery recommendations based on their activities.

Test Case Description: As a user, I want to receive personalized recovery recommendations based on my activities.

Test Suite: Rest and Recovery Tracking

Test Priority: Medium

Preconditions:

1. User is logged in.
2. User has tracked their physical activity for at least one week.

Test Data: User's activity data and rest and recovery logs.

Test Steps:

1. Navigate to the rest and recovery tracking feature.
2. Check for the presence of recovery recommendations.

Postconditions:

1. Recovery recommendations are displayed based on the user's activity data and rest and recovery logs.

Expected Result: Users receive accurate and personalized recovery recommendations based on their activities.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 4:

Test Case ID: TRR-004

Test Case Title: Verify that the app provides alerts for insufficient recovery time.

Test Case Description: As a user, I want to be alerted if I don't allow for sufficient recovery time between intense training sessions.

Test Suite: Rest and Recovery Tracking

Test Priority: Medium

Preconditions:

1. User is logged in.
2. User has tracked intense training sessions for at least one week.

Test Data: User's activity data and rest and recovery logs.

Test Steps:

1. Navigate to the rest and recovery tracking feature.
2. Check for the presence of alerts or notifications related to recovery time.

Postconditions:

1. Alerts or notifications related to recovery time are displayed if necessary.

Expected Result: The app provides alerts for insufficient recovery time between intense training sessions.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case 5:

Test Case ID: TRR-005

Test Case Title: Verify that users can view trends and history of their recovery data.

Test Case Description: As a user, I want to be able to view trends and history of my recovery data to monitor progress over time.

Test Suite: Rest and Recovery Tracking

Test Priority: High

Preconditions:

- 1. User is logged in.
- 2. User has tracked their rest and recovery data for at least one month.

Test Data: User's rest and recovery logs and metrics.

Test Steps:

- 1. Navigate to the rest and recovery tracking feature.
- 2. Use the filter or date range options to view recovery data over a specified time period.

Postconditions:

- 1. Recovery data trends and history are displayed for the specified time period.

Expected Result: Users can view and analyze trends and history of their recovery data over time.

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 ID: TC001

Test Case Title: Verify that users can access a variety of audio-guided workouts

Test Case Description: This test case aims to validate that users have the ability to access and select from a diverse range of audio-guided workouts within the Fitness Tracker App.

Test Suite: Audio Coaching Module

Test Priority: High

Preconditions:

- User must be logged in
- User must have an internet connection

Test Data: No test data needed

#### Test Steps:

1. Launch the Fitness Tracker App and log in using valid credentials.
2. Tap on the "Workouts" tab from the bottom navigation menu.
3. Select an audio-guided workout category (e.g., Strength Training, Yoga, HIIT).
4. Choose a specific workout from the available list.

#### Postconditions:

- User must be redirected to the chosen audio-guided workout

#### Expected Result:

- User can access and select from a variety of audio-guided workouts from different categories.

Severity: Major

Priority: High

Type of Testing: Functional Testing, End-to-End Testing

Test Case Approach: Positive

Test Case ID: TC002

Test Case Title: Verify that audio coaching includes motivational messages and coaching cues

Test Case Description: This test case aims to ensure that audio coaching provides users with appropriate motivational messages and coaching cues during workouts.

Test Suite: Audio Coaching Module

Test Priority: High

#### Preconditions:

- User must have an active audio-guided workout

Test Data: No test data needed

#### Test Steps:

1. Start an audio-guided workout.
2. Listen for motivational messages and coaching cues throughout the workout.
3. Document the frequency and appropriateness of the messages and cues.



Postconditions:

- User has completed the workout

Expected Result:

- Audio coaching includes motivational messages and coaching cues that are both frequent and appropriate.

Severity: Major

Priority: High

Type of Testing: Functional Testing, End-to-End Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users can choose coaching based on workout type and intensity

Test Case Description: This test case aims to ensure that users can select audio coaching based on the workout type and intensity they have chosen.

Test Suite: Audio Coaching Module

Test Priority: High

Preconditions:

- User must have an internet connection

Test Data: Different workout types and intensities

Test Steps:

1. Log in to the Fitness Tracker App.
2. Navigate to the "Workouts" section and choose a workout type (e.g., Yoga, Pilates, HIIT).
3. Select the desired workout intensity (e.g., Low, Moderate, High).
4. Start the audio-guided workout.
5. Listen for coaching cues relevant to the chosen workout type and intensity.

Postconditions:

- User has started the audio-guided workout

Expected Result:

- Audio coaching provides cues relevant to the selected workout type and intensity.

Severity: Major

Priority: High

Type of Testing: Functional Testing, End-to-End Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that audio coaching integrates seamlessly with other app features

Test Case Description: This test case aims to verify that the audio coaching feature integrates flawlessly with the other functionalities of the Fitness Tracker App.

Test Suite: Audio Coaching Module

Test Priority: High

Preconditions:

- User must log in
- User must have an active audio-guided workout

Test Data: No test data needed

Test Steps:

1. Log in and start an audio-guided workout.
2. Access the tracker for heart rate, calories burned, distance, etc., during the workout.
3. Verify that the tracker updates in real-time along with the audio coaching.
4. Pause/Resume the workout via the audio interface and observe the updated input from the tracker features.

Postconditions:

- User has completed the workout

Expected Result:

- Audio coaching integrates seamlessly with the tracking and other app features updates in sync.

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 control audio settings and preferences

Test Case Description: This test case aims to ensure that users can easily manage audio settings and preferences during an audio-guided workout.

Test Suite: Audio Coaching Module

Test Priority: Medium

Preconditions:

- User must have an active audio-guided workout

Test Data: No test data needed

Test Steps:

1. Start an audio-guided workout.
2. Tap the audio interface to access audio settings and preferences (e.g. volume, voice, pace).
3. Explore the available options and modify them as desired.
4. Observe the impact of the changes on the audio coaching.

Postconditions:

- User can navigate and manage audio settings and preferences

Expected Result:

- Users can intuitively control audio settings and preferences during an audio-guided workout.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, End-to-End 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 -**

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-001

Test Case Title: Verify that the app integrates with popular health and wellness apps

Test Case Description: Verify that the fitness tracker app can connect and synchronize data with popular health and wellness apps such as MyFitnessPal, Headspace, and Calm.

Test Suite: Integration Testing

Test Priority: High

Preconditions:

- User has an account in the fitness tracker app
- User has internet connectivity
- User has accounts in MyFitnessPal, Headspace, and Calm

Test Data: User credentials for the fitness tracker app and health apps

Test Steps:

1. Launch the fitness tracker app
2. Navigate to the app settings
3. Select the "Integrations" option
4. Search for and select MyFitnessPal, Headspace, and Calm from the list of available apps
5. Enter user credentials for each app to establish a connection
6. Synchronize data between the fitness tracker app and health apps

Postconditions:

- Fitness tracker app is connected to MyFitnessPal, Headspace, and Calm
- User data from health apps is visible in the fitness tracker app

Expected Result: Successful integration with popular health and wellness apps, with user data from health apps displayed in the fitness tracker app

Severity: Major

Priority: High

Type of Testing: Integration Testing

Test Case Approach: Positive

Test Case ID: TC-002

Test Case Title: Verify that users can sync data from connected health apps

Test Case Description: Verify that users can manually sync data from connected health and wellness apps within the fitness tracker app.

Test Suite: Integration Testing

Test Priority: Medium

Preconditions:

- User has an account in the fitness tracker app
- User has internet connectivity
- User is connected to MyFitnessPal, Headspace, and Calm

Test Data: No test data needed

Test Steps:

1. Launch the fitness tracker app
2. Navigate to the "Connected Apps" section in the app settings
3. Select "Sync Data" for each connected app (MyFitnessPal, Headspace, and Calm)
4. Wait for the data synchronization process to complete

Postconditions:

- Data from health apps is updated in the fitness tracker app

Expected Result: Users can manually synchronize data from connected health and wellness apps, with updated user data displayed in the fitness tracker app

Severity: Major

Priority: Medium

Type of Testing: Integration Testing

Test Case Approach: Positive

Test Case ID: TC-003

Test Case Title: Verify that the app provides a consolidated view of health metrics

Test Case Description: Verify that the fitness tracker app combines health metrics from connected apps into a

single, easily understood dashboard.

Test Suite: Functional Testing

Test Priority: High

Preconditions:

- User has an account in the fitness tracker app
- User has internet connectivity
- User is connected to MyFitnessPal, Headspace, and Calm

Test Data: No test data needed

Test Steps:

1. Launch the fitness tracker app
2. Navigate to the main dashboard
3. Verify that health metrics such as steps taken, calories burned, distance traveled, and heart rate from all connected apps are displayed

Postconditions:

- Health metrics from all connected apps are visible on the main dashboard

Expected Result: The fitness tracker app provides a consolidated view of health metrics from connected apps on the main dashboard

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-004

Test Case Title: Verify that users receive insights based on combined data from multiple apps

Test Case Description: Verify that users receive actionable insights and personalized recommendations based on the combined data from all connected health and wellness apps.

Test Suite: Functional Testing

Test Priority: High

Preconditions:

- User has an account in the fitness tracker app
- User has internet connectivity
- User is connected to MyFitnessPal, Headspace, and Calm
- User has tracked health data in connected apps

Test Data: User data from health apps

Test Steps:

1. Launch the fitness tracker app
2. Navigate to the "Insights" section
3. Verify that insights and recommendations are generated based on the user's combined health data

Postconditions:

- Actionable insights and personalized recommendations are provided to the user

Expected Result: Users receive insights and recommendations based on the combined data from all connected health and wellness apps

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-005

Test Case Title: Verify that integration settings are easy to manage and configure

Test Case Description: Verify that users can easily manage and configure their integration settings on the fitness tracker app, enabling or disabling data synchronization from connected apps.

Test Suite: Functional Testing

Test Priority: Medium

Preconditions:



- User has an account in the fitness tracker app
- User has internet connectivity
- User is connected to MyFitnessPal, Headspace, and Calm

Test Data: No test data needed

Test Steps:

1. Launch the fitness tracker app
2. Navigate to the "App Settings"
3. Select the "Connected Apps" option
4. Enable or disable data synchronization for each connected app (MyFitnessPal, Headspace, and Calm)

Postconditions:

- Data synchronization settings can be managed for each connected app

Expected Result: Users can easily manage and configure their integration settings on the fitness tracker app, enabling or disabling data synchronization from connected apps

Severity: Major

Priority: Medium

Type of Testing: Functional 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 ID: TC001

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 Fitness Tracker App allows users to log their body composition metrics including weight, body fat percentage, and muscle mass.

Test Suite: Body Composition Tracking

Test Priority: High

Preconditions:

- User is logged into the app
- User has access to the Body Composition Tracking feature

Test Data: User data (weight, body fat percentage, muscle mass)

Test Steps:

1. Navigate to the Body Composition Tracking feature
2. Select the 'Log Data' option
3. Enter user data for weight, body fat percentage, and muscle mass
4. Save the entered data

Postconditions:

- Entered data is saved and displayed in the user's body composition history

Expected Result: User is able to log body composition metrics successfully.

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 BMI based on logged data.

Test Case Description: This test case confirms that the Fitness Tracker App correctly calculates and displays BMI based on the logged weight and height data.

Test Suite: Body Composition Tracking

Test Priority: High

Preconditions:

- User is logged into the app
- User has access to the Body Composition Tracking feature
- User has logged weight and height data

Test Data: Stored user data (weight and height)

Test Steps:

1. Navigate to the Body Composition Tracking feature

2. View the user's logged data

3. Verify that the app calculates and displays the BMI based on the logged weight and height

Postconditions:

- BMI is accurately calculated and displayed based on the user's logged data

Expected Result: The app correctly calculates and displays the user's BMI.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that users can view trends and history of body composition changes.

Test Case Description: This test case ensures that the Fitness Tracker App allows users to view trends and history of their body composition changes over time.

Test Suite: Body Composition Tracking

Test Priority: Medium

Preconditions:

- User is logged into the app
- User has access to the Body Composition Tracking feature
- User has logged body composition metrics for at least one week

Test Data: Stored user data (body composition metrics)

Test Steps:

1. Navigate to the Body Composition Tracking feature
2. Select the 'History' or 'Trends' option
3. Verify that the app displays the user's body composition history and trends for the selected duration

Postconditions:

- User is able to view their body composition history and trends over time

Expected Result: Users can view trends and history of their body composition changes.

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 insights and recommendations based on body composition data.

Test Case Description: This test case verifies that the Fitness Tracker App generates insights and recommendations based on the user's body composition data.

Test Suite: Body Composition Tracking

Test Priority: Medium

Preconditions:

- User is logged into the app
- User has access to the Body Composition Tracking feature
- User has logged body composition metrics for at least one month

Test Data: Stored user body composition data

Test Steps:

1. Navigate to the Body Composition Tracking feature
2. Select the 'Insights' or 'Recommendations' option
3. Verify that the app generates insights and recommendations based on the user's body composition data

Postconditions:

- App provides accurate insights and recommendations based on user body composition data

Expected Result: The app generates insights and recommendations based on the user's body composition data.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that users can set goals for body composition metrics.

Test Case Description: This test case confirms that the Fitness Tracker App allows users to set goals for various body composition metrics.

Test Suite: Body Composition Tracking

Test Priority: Medium

Preconditions:

- User is logged into the app
- User has access to the Body Composition Tracking feature

Test Data: Body composition goal values

Test Steps:

1. Navigate to the Body Composition Tracking feature
2. Select the 'Set Goals' option
3. Enter desired goal values for body composition metrics (weight, body fat percentage, muscle mass, BMI)
4. Save the entered goal values

Postconditions:

- Goal values are saved and displayed in the user's body composition goal section

Expected Result: User is able to set goals for body composition metrics successfully.

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 -**

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 ID: FTW-001

Test Case Title: Verify that users can view detailed performance metrics for each workout

Test Case Description: This test case ensures that the Fitness Tracker App correctly displays workout performance metrics, including steps taken, distance traveled, and calories burned.

Test Suite: Workout Analysis

Test Priority: High

Preconditions:

- User has an active account
- User has completed at least one workout
- Workout data is available for the user

Test Data: User data with a completed workout

Test Steps:

1. Log in to the Fitness Tracker App
2. Navigate to the "Workouts" section
3. Select a completed workout from the list
4. View the performance metrics for the selected workout

Postconditions:

- Performance metrics are displayed on the screen
- User can navigate back to the workout list

Expected Result: The app displays detailed performance metrics for the selected workout, including steps taken, distance traveled, and calories burned.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: FTW-002

Test Case Title: Verify that the app provides trends and insights into workout intensity and duration

Test Case Description: This test case checks if the Fitness Tracker App identifies patterns and displays insights about users' workout intensity and duration, helping them assess their performance and optimize



training strategies.

Test Suite: Workout Analysis

Test Priority: High

Preconditions:

- User has completed multiple workouts
- Workout data is available for the user

Test Data: User data with multiple workouts

Test Steps:

1. Log in to the Fitness Tracker App
2. Navigate to the "Workouts" section
3. Select "Trends & Insights" from the menu
4. Analyze the displayed trends and insights related to workout intensity and duration

Postconditions:

- Trends and insights are displayed on the screen
- User can navigate back to the workout list

Expected Result: The app provides trends and insights into workout intensity and duration, highlighting patterns and suggesting areas for improvement or optimization.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: FTW-003

Test Case Title: Verify that users can track calorie burn and other health metrics

Test Case Description: This test case verifies that the Fitness Tracker App accurately records and displays

calorie burn and other essential health metrics for users' workouts.

Test Suite: Workout Analysis

Test Priority: High

Preconditions:

- User has completed at least one workout
- Workout data is available for the user

Test Data: User data with a completed workout

Test Steps:

1. Log in to the Fitness Tracker App
2. Navigate to the "Workouts" section
3. Select a completed workout from the list
4. View the health metrics for the selected workout

Postconditions:

- Health metrics are displayed on the screen
- User can navigate back to the workout list

Expected Result: The app displays calorie burn and other health metrics for the selected workout, including heart rate, active minutes, etc.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: FTW-004

Test Case Title: Verify that the app offers recommendations based on workout analysis

Test Case Description: This test case validates that the Fitness Tracker App suggests personalized

recommendations based on users' workout analysis, assisting them in optimizing their training strategies.

Test Suite: Workout Analysis

Test Priority: High

Preconditions:

- User has completed multiple workouts
- Workout data is available for the user

Test Data: User data with multiple workouts

Test Steps:

1. Log in to the Fitness Tracker App
2. Navigate to the "Workouts" section
3. Select "Trends & Insights" from the menu
4. Review the displayed recommendations related to workout optimization

Postconditions:

- Recommendations are displayed on the screen
- User can navigate back to the workout list

Expected Result: The app provides personalized recommendations based on users' workout analysis, suggesting areas for improvement, progress tracking, and training optimization.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: FTW-005

Test Case Title: Verify that users can compare current workouts with past performance

Test Case Description: This test case confirms that the Fitness Tracker App allows users to compare their

current workout performance with past achievements, encouraging them to set goals and stay motivated.

Test Suite: Workout Analysis

Test Priority: High

Preconditions:

- User has completed multiple workouts
- Workout data is available for the user

Test Data: User data with multiple workouts

Test Steps:

1. Log in to the Fitness Tracker App
2. Navigate to the "Workouts" section
3. Select a completed workout from the list
4. Compare the selected workout with previous workout data

Postconditions:

- Comparison is displayed on the screen
- User can navigate back to the workout list

Expected Result: The app allows users to compare their current workout performance with past achievements, highlighting improvements and encouraging them to set goals and stay motivated.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

-----

**Data - 97**

**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 ID: TC001

Test Case Title: Verify that the app provides current weather conditions and forecasts

Test Case Description: As a user, I want to verify that the app displays accurate current weather conditions and forecasts for my location.

Test Suite: Weather Integration

Test Priority: High

Preconditions:

- User is registered and logged in

- User has granted location permissions to the app

Test Data: User's current location

Test Steps:

1. Open the Fitness Tracker App
2. Log in to the app with valid user credentials
3. Grant location permissions if prompted
4. Navigate to the Weather Forecast section
5. Check the current weather conditions (temperature, humidity, wind speed, etc.)
6. Scroll down to view the weather forecast for upcoming days

Postconditions:

- The app displays the user's current location
- The app presents accurate current weather conditions
- The app shows a forecast for the next few days

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 ID: TC002

Test Case Title: Verify that users can view weather details for different locations

Test Case Description: As a user, I want to verify that I can view weather details for different locations within the app.

Test Suite: Weather Integration

Test Priority: Medium

Preconditions:

- User is registered and logged in

Test Data: Multiple location coordinates

Test Steps:

1. Open the Fitness Tracker App
2. Log in to the app with valid user credentials
3. Navigate to the Weather Forecast section
4. Select "Change Location" or a similar option
5. Enter a different location's address, city, or coordinates
6. Confirm the selected location
7. Verify that the app displays accurate current weather conditions and a forecast for the new location

Postconditions:

- The app displays the new location's weather information

Expected Result:

- The app allows users to view weather details for different locations, accurately displaying current conditions and forecasts.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing, System Testing

Test Case Approach: Positive

Test Case ID: TC003

Test Case Title: Verify that weather data is updated in real-time

Test Case Description: As a user, I want to verify that the app updates weather data in real-time to ensure accuracy.

Test Suite: Weather Integration

Test Priority: High

Preconditions:

- User is registered and logged in
- User has granted location permissions to the app

Test Data: No test data needed

Test Steps:

1. Open the Fitness Tracker App
2. Log in to the app with valid user credentials
3. Grant location permissions if prompted
4. Navigate to the Weather Forecast section
5. Verify the current weather conditions (temperature, humidity, wind speed, etc.)
6. Wait for approximately 15 minutes
7. Refresh the Weather Forecast section
8. Verify if any changes are present in the current weather conditions

Postconditions:

- The app updates current weather conditions when necessary

Expected Result:

- The app updates weather data in real-time, reflecting any changes in the user's current weather conditions.

Severity: Major

Priority: High

Type of Testing: Functional Testing, System Testing

Test Case Approach: Positive

Test Case ID: TC004

Test Case Title: Verify that the app offers recommendations based on weather conditions

Test Case Description: As a user, I want to verify that the app provides recommendations based on weather conditions to optimize my outdoor workouts.

Test Suite: Weather Integration, Recommendation System

Test Priority: High



Preconditions:

- User is registered and logged in
- User has granted location permissions to the app

Test Data: User's current location, user's preferred activities, weather data

Test Steps:

1. Open the Fitness Tracker App
2. Log in to the app with valid user credentials
3. Grant location permissions if prompted
4. Navigate to the Recommendation section
5. Verify if any recommendations are provided based on the current weather conditions

Postconditions:

- The app presents recommendations based on weather conditions

Expected Result:

- The app recommends suitable outdoor activities based on the user's current weather conditions.

Severity: Major

Priority: High

Type of Testing: Functional Testing, System Testing, Integration Testing

Test Case Approach: Positive

Test Case ID: TC005

Test Case Title: Verify that users receive weather alerts for extreme conditions

Test Case Description: As a user, I want to verify that the app sends weather alerts for extreme conditions to ensure my safety.

Test Suite: Weather Integration, Notification System

Test Priority: High

Preconditions:

- User is registered and logged in

- User has granted location permissions and notification permissions to the app

Test Data: Extreme weather conditions (e.g., tornado warning)

Test Steps:

1. Open the Fitness Tracker App
2. Log in to the app with valid user credentials
3. Grant location permissions and notification permissions if prompted
4. Leave the app running in the background
5. Trigger an extreme weather condition alert in the test environment

Postconditions:

- The app sends a weather alert notification to the user

Expected Result:

- The app delivers weather alerts for extreme conditions, notifying users proactively.

Severity: Critical

Priority: High

Type of Testing: Functional Testing, System Testing, Integration Testing

Test Case Approach: Positive

-----

**Data - 98**

**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 ID: TC-001

Test Case Title: Verify that users can log their mood, energy levels, and motivation.

Test Suite: Mood Tracking Input

Test Priority: High

Preconditions: User is logged in, user has access to the mood tracking feature

Test Data: Predefined mood values, energy levels, and motivation inputs

Test Steps:

1. Navigate to the mood tracking feature.
2. Select a mood value from the list (e.g. happy, neutral, sad).
3. Input a value for energy levels (e.g. high, medium, low).
4. Input a value for motivation (e.g. high, medium, low).

5. Save the input data.

Postconditions: Mood, energy levels, and motivation data are saved and displayed in the user's history

Expected Result: Mood, energy levels, and motivation data are saved and accurately displayed in the user's history

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-002

Test Case Title: Verify that users receive recommendations based on their mood and energy data.

Test Suite: Mood Tracking Recommendations

Test Priority: High

Preconditions: User has logged mood, energy levels, and motivation data

Test Data: Predefined user data, exercise recommendations database

Test Steps:

1. Navigate to the recommendations feature.
2. Analyze the user's logged mood, energy levels, and motivation data.
3. Compare the user data with the exercise recommendations database.
4. Display tailored exercise recommendations based on the user data.

Postconditions: Recommendations are displayed and stored in the user's history

Expected Result: Tailored exercise recommendations are displayed and stored based on the user's logged mood, energy levels, and motivation data

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-003

Test Case Title: Verify that the app supports journaling for personal reflections on mood and workouts.

Test Suite: Mood Tracking Journaling

Test Priority: Medium

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. Input a text entry reflecting on the user's mood, energy levels, or workout experience.
3. Save the journal entry.
4. View the saved journal entry.

Postconditions: Journal entry is saved and displayed in the user's history

Expected Result: Journal entries are saved and accurately displayed in the user's history

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-004

Test Case Title: Verify that users can set goals for improving mood and energy levels.

Test Suite: Mood Tracking Goals

Test Priority: Medium

Preconditions: User is logged in, user has access to the goal-setting feature

Test Data: Predefined goal inputs

Test Steps:

1. Navigate to the goal-setting feature.

2. Input a goal for improving mood (e.g. "Feel happier during workouts").
3. Input a goal for improving energy levels (e.g. "Increase energy levels before workouts").
4. Set a target date for achieving the goals.
5. Save the goal inputs.

Postconditions: Goals are saved and displayed in the user's dashboard

Expected Result: Goals are saved and accurately displayed in the user's dashboard

Severity: Minor

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-005

Test Case Title: Verify that the app provides insights into mood and energy trends over time.

Test Suite: Mood Tracking Data Analysis

Test Priority: High

Preconditions: User has logged mood, energy levels, and motivation data over a period of time

Test Data: Predefined user data, data analysis algorithms

Test Steps:

1. Navigate to the data analysis feature.
2. Analyze the user's logged mood, energy levels, and motivation data.
3. Identify trends in the user's data.
4. Display visualizations (e.g. charts, graphs) illustrating the trends.

Postconditions: Trends are identified and displayed in visualizations

Expected Result: Accurate trends in the user's mood and energy levels are identified and displayed in visualizations

Severity: Major

Priority: High

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 ID: TC001

Test Case Title: Verify that users can input and save ingredients for a recipe

Test Case Description: This test case verifies that users are able to add, edit, and save the list of ingredients for their recipes.

Test Suite: Recipe Creation

Test Priority: High

Preconditions:

- User is logged into the platform
- User is on the recipe creation page

Test Data: Any valid recipe ingredient(s)

Test Steps:

1. Access the recipe creation page
2. Input a valid recipe ingredient in the 'Ingredients' text field
3. Click the 'Add Ingredient' button
4. Repeat steps 2 and 3 for multiple ingredients
5. Edit an existing ingredient
6. Save the recipe

Postconditions:

- Ingredients are saved and displayed on the recipe details page

Expected Result: Recipe contains the correct ingredients, and updates are saved 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 input and save cooking instructions for a recipe

Test Case Description: This test case verifies that users are able to add, edit, and save cooking instructions for their recipes.

Test Suite: Recipe Creation

Test Priority: High

Preconditions:

- User is logged into the platform
- User is on the recipe creation page

Test Data: Any valid cooking instruction(s)

Test Steps:

1. Access the recipe creation page
2. Input a valid cooking instruction in the 'Instructions' text area
3. Click the 'Add Instruction' button
4. Repeat steps 2 and 3 for multiple instructions
5. Edit an existing instruction
6. Save the recipe

Postconditions:

- Cooking instructions are saved and displayed on the recipe details page

Expected Result: Recipe contains the correct cooking instructions, and updates are saved 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 upload photos for their recipes

Test Case Description: This test case verifies that users are able to upload photos to accompany their recipes.

Test Suite: Recipe Creation

Test Priority: High

Preconditions:

- User is logged into the platform
- User is on the recipe creation page

Test Data: A valid image file (e.g., .jpg, .png)

Test Steps:

1. Access the recipe creation page
2. Click the 'Upload Photo' button
3. Select a valid image file from the system
4. Wait for the image to upload and display a preview
5. Save the recipe

Postconditions:

- The uploaded photo is saved and displayed on the recipe details page

Expected Result: The recipe has the correct image, and the upload is successful.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC004

Test Case Title: Verify that recipes are saved in the user's profile

Test Case Description: This test case verifies that recipes created by a user are saved in their profile and accessible.

Test Suite: Recipe Creation

Test Priority: High

Preconditions:

- User is logged into the platform
- User has created a recipe

Test Data: No test data needed

Test Steps:

1. Log in as the user who created the recipe
2. Navigate to the user's profile page
3. Locate the saved recipe on the user's profile page

Postconditions:

- The recipe is visible and accessible on the user's profile page

Expected Result: The recipe appears in the user's profile and can be accessed by other users on the platform.

Severity: Critical

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

---

Test Case ID: TC005

Test Case Title: Verify that users receive a confirmation upon successful recipe submission

Test Case Description: This test case verifies that users receive visual or auditory confirmation after successfully submitting a recipe.

Test Suite: Recipe Creation

Test Priority: Medium

Preconditions:

- User is logged into the platform
- User is on the recipe creation page

Test Data: No test data needed

Test Steps:

1. Access the recipe creation page
2. Fill out all required fields for the recipe
3. Click the 'Submit' or 'Save' button to create the recipe

Postconditions:

- The recipe is saved and appears in the user's profile
- Visible or audible confirmation is displayed (e.g., success message or animation)

Expected Result: The recipe is saved, and the user receives visual or auditory confirmation of successful submission.

Severity: Major

Priority: Medium

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 aims to validate that the recipe sharing platform allows users to search for recipes using keywords and returns relevant results.

Test Suite: Recipe Discovery

Test Priority: High

Preconditions:

- User is logged in or can search as a guest.
- Recipes with various keywords have been added to the system.

Test Data: Keywords related to recipes, e.g., "pasta", "chocolate cake", "vegetarian"

#### Test Steps:

1. Navigate to the recipe search page.
2. Enter a keyword in the search bar.
3. Click the search button.
4. Review the search results.

#### Postconditions:

- Search results are displayed.
- Recipes with matching keywords are included in the results.

Expected Result: Recipe search by keyword returns relevant results.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-RD-002

Test Case Title: Verify that users can search for recipes using ingredients

Test Case Description: This test case validates that the recipe sharing platform allows users to search for recipes using ingredients and returns relevant results.

Test Suite: Recipe Discovery

Test Priority: High

#### Preconditions:

- User is logged in or can search as a guest.
- Recipes with various ingredients have been added to the system.

Test Data: Ingredients related to recipes, e.g., "garlic", "tomatoes", "chicken"

#### Test Steps:

1. Navigate to the recipe search page.
2. Enter an ingredient in the search bar.

3. Click the search button.

4. Review the search results.

Postconditions:

- Search results are displayed.
- Recipes with matching ingredients are included in the results.

Expected Result: Recipe search by ingredient returns relevant results.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-RD-003

Test Case Title: Verify that search results accurately match search criteria

Test Case Description: This test case aims to confirm that the recipe sharing platform returns search results that accurately match the user's search criteria.

Test Suite: Recipe Discovery

Test Priority: High

Preconditions:

- User is logged in or can search as a guest.
- Recipes with various search criteria (keywords, ingredients, cuisine, dietary preferences, etc.) have been added to the system.

Test Data: Keywords, ingredients, and cuisine preferences

Test Steps:

1. Navigate to the recipe search page.
2. Enter a keyword and/or ingredient and select cuisine type or dietary preference.
3. Click the search button.
4. Review the search results for accuracy based on entered criteria.

Postconditions:

- Search results are displayed.
- Recipes match the entered search criteria accurately.

Expected Result: Search results match user's search criteria accurately.

Severity: Major

Priority: High

Type of Testing: Functional Testing

Test Case Approach: Positive

Test Case ID: TC-RD-004

Test Case Title: Verify that recipes in search results are accessible and detailed

Test Case Description: This test case aims to confirm that the recipe sharing platform displays complete and detailed recipes in the search results.

Test Suite: Recipe Discovery

Test Priority: Medium

Preconditions:

- User is logged in or can search as a guest.
- Recipes with various ingredients, instructions, and images have been added to the system.

Test Steps:

1. Navigate to the recipe search page.
2. Perform a search using keywords or ingredients.
3. Review the search results.
4. Click on a recipe in the search results.
5. Verify that the recipe details are comprehensive, including ingredients, instructions, and an image.

Postconditions:

- Recipe details are displayed.
- Recipe information is complete and detailed.



Expected Result: Recipe details are accessible and comprehensive in the search results.

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 confirms that the recipe sharing platform allows users to save search results for future reference.

Test Suite: Recipe Discovery

Test Priority: Medium

Preconditions:

- User is logged in.
- Recipes have been added to the system.

Test Steps:

1. Navigate to the recipe search page.
2. Perform a search using keywords or ingredients.
3. Review the search results.
4. Save the search results.
5. Log out and log back in.
6. View saved search results.

Postconditions:

- Saved search results are available for the user.

Expected Result: Users can save and access search results for future reference.

Severity: Major

Priority: Medium

Type of Testing: Functional Testing

Test Case Approach: Positive

-----