

# Test Plan for Kraken

Group 12

Alex Liu, Youssef Ismail, Elena Arnold

Name	Last Modified	Reason for Changes	Version
Elena Arnold Youssef Ismail Alex Liu	Apr 5, 2024	Initial version	1.0

## Objective

The objective of this test plan is to demonstrate Kraken's conformance with the functional requirements stated in the software requirement specifications (SRS). The test plan aims to cover all functional requirements in the SRS. Functionalities in both front and back end will be tested to ensure their correctness and compliance.

## Testing Techniques

For our testing plan, we are planning to use black box testing in frontend to validate the implemented features according to the SRS. We will conduct end-to-end testing to evaluate the system's behavior across various modules and components. The real world usage of the application will be simulated by conducting system-level acceptance tests. This ensures that the software delivers the intended functionality for the users.

For backend, white box testing is used to ensure the code implementation meets the design requirements. Backend code is tested on its correctness of handling API requests and database operations.

## Testing Coverage

For frontend testing, the tests are to be designed based on the equivalence partitioning coverage, where positive testing and negative testing are employed to test if the system performs its intended function correctly and handles errors/exceptions gracefully.

For backend testing, we strive for 100% statement coverage for the code implementation, through unit and system testing.

## Testing Instruction

For frontend testing, the test cases can be manually executed by following the test steps.

For backend testing:

1. Navigate to the root directory
2. Run `python3 -m unittest`
3. To run a specific file run you can specify it like so `python3 -m unittest api/tests/test_projects.py`

For generating coverage report for backend:

1. Run `pip install coverage` to install the library
2. Run `python -m coverage run --source=api -m unittest`
3. Run `python -m coverage report > coverage.txt` to save the coverage report to a file.
4. Run `python -m coverage html` to generate a html report.

## Frontend Tests

Name	Last Modified	Reason for Changes	Version
Elena Arnold Youssef Ismail Alex Liu	Mar 29, 2024	Initial version	1.0
Elena Arnold Youssef Ismail Alex Liu	Apr 5, 2024	Review and rerun tests	1.1

Total Test Cases	Last Executed	Passed Test Cases	Pass %
69	Apr 5, 2024	69	100%

# FR1: Request Registration

Priority	High ▾
Description	This test case verifies users are able to register with the system
Test Objective	FR1: Request.Registration

Test Case Id	1
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Open the Home Page	<input type="checkbox"/>
Expected Result: Home page is displayed	
2. Click on the Sign Up button	<input type="checkbox"/>
Expected Result: Sign up form is displayed	
3. Enter a unique email, username, and password and click the Create Account button	<input type="checkbox"/>
Expected Result: User is redirected to Login page and success toast message appears	
Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

<b>Priority</b>	Medium ▾
<b>Description</b>	This test case verifies error checking exists for the fields on the Registration page
<b>Test Objective</b>	FR1: Request.Registration

<b>Test Case Id</b>	2
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Open the Home Page</b>	<input type="checkbox"/>
Expected Result: Home page is displayed	
<b>2. Click on the Sign Up button</b>	<input type="checkbox"/>
Expected Result: Sign up form is displayed	
<b>3. Click on the Username field and click off the Username field</b>	<input type="checkbox"/>
Expected Result: ‘Username required’ error message is displayed	
<b>4. Click on the Email Address field and click off the Email Address field</b>	<input type="checkbox"/>
Expected Result: ‘Email Address required’ error message is displayed	
<b>5. Enter ‘email’ in the Email Address field and click Create Account</b>	<input type="checkbox"/>
Expected Result: ‘Please include an ‘@’ in the email address. ‘Email’ is missing and ‘@’.’ error message is displayed	
<b>6. Click on the Password field and click off the Password field</b>	<input type="checkbox"/>
Expected Result: ‘Password required’ error message is displayed	
<b>7. Click on the Password field and click off the Password field</b>	<input type="checkbox"/>
Expected Result: ‘Password required’ error message is displayed	
<b>8. Click on the Password field, enter ‘x’, and click off the Password field</b>	<input type="checkbox"/>

Expected Result: ‘Password must be at least 8 characters’ error message is displayed

<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

<b>Priority</b>	High <input type="button" value="▼"/>
<b>Description</b>	This test case verifies users cannot register an account with an username that already exists in the system
<b>Test Objective</b>	FR1: Request.Registration

<b>Test Case Id</b>	3
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Open the Home Page</b>	<input type="checkbox"/>
Expected Result: Home page is displayed	
<b>2. Click on the Sign Up button</b>	<input type="checkbox"/>
Expected Result: Sign up form is displayed	
<b>3. Enter:</b>  <b>Username:</b> John Doe <b>Email Address:</b> jdoe@email.com <b>Password:</b> Any Expected Result: Error toast message is displayed: "Username already exists"	<input type="checkbox"/>
<b>Comment</b>	
<b>Execution status</b>	Pass <input type="button" value="▼"/>
<b>Bug Severity</b>	None <input type="button" value="▼"/>
<b>Bug Priority</b>	None <input type="button" value="▼"/>

<b>Priority</b>	Low ▾
<b>Description</b>	This test case verifies users are able to navigate back to the home page from the registration page
<b>Test Objective</b>	FR1: Request.Registration

<b>Test Case Id</b>	4
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Open the Home Page</b>	<input type="checkbox"/>
Expected Result: Home page is displayed	
<b>2. Click on the Sign Up button</b>	<input type="checkbox"/>
Expected Result: Registration page is displayed	
<b>3. Click on the Back button</b>	<input type="checkbox"/>
Expected Result: Home page is displayed	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR2: Request.Login

Priority	High ▾
Description	This test case verifies users are able to login to the system
Test Objective	FR2: Request.Login

Test Case Id	5
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Open the Home Page	<input type="checkbox"/>
Expected Result: Home page is displayed	
2. Click on the Login button	<input type="checkbox"/>
Expected Result: Login form is displayed	
3. Enter valid credentials and click the Login button	<input type="checkbox"/>
Expected Result: Projects page is displayed	
Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

<b>Priority</b>	Medium ▾
<b>Description</b>	This test case verifies error checking exists for the fields on the Login page
<b>Test Objective</b>	FR2: Request.Login

<b>Test Case Id</b>	6
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>Open the Home Page</b>	<input type="checkbox"/>
Expected Result: Home page is displayed	
<b>Click on the Login button</b>	<input type="checkbox"/>
Expected Result: Sign up form is displayed	
<b>Click on the Username field and click off the Username field</b>	<input type="checkbox"/>
Expected Result: ‘Username required’ error message is displayed	
<b>Click on the Password field and click off the Password field</b>	<input type="checkbox"/>
Expected Result: ‘Password required’ error message is displayed	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

<b>Priority</b>	Low ▾
<b>Description</b>	This test case verifies users are able to navigate back to the home page from the login page
<b>Test Objective</b>	FR2: Request.Login

<b>Test Case Id</b>	7
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Open the Home Page</b>	<input type="checkbox"/>
Expected Result: Home page is displayed	
<b>2. Click on the Login button</b>	<input type="checkbox"/>
Expected Result: Login page is displayed	
<b>3. Click on the Back button</b>	<input type="checkbox"/>
Expected Result: Home page is displayed	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR3: Change.Password

Priority	High ▾
Description	This test case verifies users are able to change their password
Test Objective	FR
Prerequisites	User is logged in

Test Case Id	8
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on the User Profile button	<input type="checkbox"/>
Expected Result: Dropdown menu is displayed	
2. Click on Change Password	<input type="checkbox"/>

    Expected Result: Change password page is displayed

3. Enter new password and confirm password and click Change Password

    Expected Result: Projects page is displayed and success toast message appears

Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

<b>Priority</b>	Medium ▾
<b>Description</b>	This test case verifies error checking exists for the fields on the Change Password page
<b>Test Objective</b>	FR3: Change.Password
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	9
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

### Test Steps

- Click on the New Password field and click off the New Password field**   
Expected Result: ‘Password required’ error message is displayed
- Click on the Confirm New Password field and click off the Confirm New Password field**   
Expected Result: ‘Password verification required’ error message is displayed
- Enter ‘x’ in the New Password field and click off the New Password field**   
Expected Result: ‘Password must be at least 8 characters’ error message is displayed
- Enter ‘x’ in the Confirm New Password field and click off the Confirm New Password field**   
Expected Result: ‘Password must be at least 8 characters’ error message is displayed

<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾

Bug Priority	None ▾
--------------	--------

<b>Priority</b>	Low ▾
<b>Description</b>	This test case verifies users can navigate back to the Projects page from the Change Password page
<b>Test Objective</b>	FR3: Change.Password
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	10
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on the User Profile button</b>	<input type="checkbox"/>
Expected Result: Dropdown menu is displayed	
<b>2. Click on Change Password</b>	<input type="checkbox"/>
Expected Result: Change password page is displayed	
<b>3. Click on Cancel</b>	<input type="checkbox"/>
Expected Result: Projects page is displayed	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR4: Select Project

Priority	High ▾
Description	This test case verifies users are able to select a project on the Projects page
Test Objective	FR4: Select.Project
Prerequisites	User is logged in User must own or be a team member on a project

Test Case Id	11
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project card	<input type="checkbox"/>
Expected Result: Kanban page is displayed for that project	
Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

## FR5: Delete Project

Priority	High ▾
Description	This test case verifies a project administrator is able to delete a project
Test Objective	FR5: Delete.Project
Prerequisites	User is logged in User owns the selected project

Test Case Id	12
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project owned by the user	<input type="checkbox"/>
Expected Result: Kanban page is displayed for the project	
2. Click on the Settings button	<input type="checkbox"/>
Expected Result: Settings page is displayed	
3. Click on the Delete Project button	<input type="checkbox"/>
Expected Result: Confirm Project Deletion modal is displayed	
4. Click on the Delete button	<input type="checkbox"/>
Expected Result: Project Deleted success toast message is displayed	
Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies a project member is not able to delete a project if they are not the project administrator
<b>Test Objective</b>	FR5: Delete.Project
<b>Prerequisites</b>	User is logged in User does not own the selected project

<b>Test Case Id</b>	13
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project not owned by the user</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed for the project	
<b>2. Click on the Settings button</b>	<input type="checkbox"/>
Expected Result: Settings page is displayed. There is a Leave Project button in place of the Delete Project button	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR6: Add.Project

Priority	High ▾
Description	This test case verifies the user is able to create a project
Test Objective	FR6: Add.Project
Prerequisites	User is logged in

Test Case Id	14
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on the Add button on the Project page	<input type="checkbox"/>
Expected Result: Create Project page is displayed	
2. Enter a Project Name and (Optional) Description and click Create Project	<input type="checkbox"/>
Expected Result: Project Created success toast message is displayed. Project Code modal is displayed.	
3. Click on the Close button	<input type="checkbox"/>
Expected Result: Projects page is displayed. New project is displayed in the Projects list.	
Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

<b>Priority</b>	Medium ▾
<b>Description</b>	This test case verifies there is error checking on the fields of the Create Project page
<b>Test Objective</b>	FR6: Add.Project
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	15
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on the Add button on the Project page</b>	<input type="checkbox"/>
Expected Result: Create Project page is displayed	
<b>2. Click on the Project Name field and click off the Project Name Field</b>	<input type="checkbox"/>
Expected Result: ‘Project name required’ error message is displayed	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

<b>Priority</b>	Low ▾
<b>Description</b>	This test case verifies user are able to cancel project creation on the Create Project page
<b>Test Objective</b>	FR6: Add.Project
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	16
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on the Add button on the Project page</b>	<input type="checkbox"/>
Expected Result: Create Project page is displayed	
<b>2. Click on the Cancel button</b>	<input type="checkbox"/>
Expected Result: Projects page is displayed	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR7: Edit.Project

Priority	High ▾
Description	This test case verifies project administrators can edit the project details including the project name and description
Test Objective	FR7: Edit.Project
Prerequisites	User is logged in User owns the selected project

Test Case Id	17
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project owned by the user	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
2. Click on the Settings button	<input type="checkbox"/>
Expected Result: Settings page is displayed	
3. Click on the Edit button	<input type="checkbox"/>
Expected Result: Edit Project page is displayed	
4. Modify the Project Name and Description and click the Save button	<input type="checkbox"/>
Expected Result: Settings page is displayed. 'Project Successfully Updated' success message is displayed. Updated project name and description is displayed.	

Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

<b>Priority</b>	Medium ▾
<b>Description</b>	This test case verifies there is error checking on the fields of the Edit Project page
<b>Test Objective</b>	FR7: Edit.Project
<b>Prerequisites</b>	User is logged in User owns the selected project

<b>Test Case Id</b>	18
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project owned by the user</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Settings button</b>	<input type="checkbox"/>
Expected Result: Settings page is displayed	
<b>3. Click on the Edit button</b>	<input type="checkbox"/>
Expected Result: Edit Project page is displayed	
<b>4. Remove the Project Name and click off the Project Name field</b>	<input type="checkbox"/>
Expected Result: ‘Project name required’ error message is displayed.	

<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies project members cannot edit the project if they are not the project administrator
<b>Test Objective</b>	FR7: Edit.Project
<b>Prerequisites</b>	User is logged in User does not own the selected project

<b>Test Case Id</b>	19
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project not owned by the user</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Settings button</b>	<input type="checkbox"/>
Expected Result: Settings page is displayed. There is no edit button displayed by the project name.	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR8: Join.Project

Priority	High ▾
Description	This test case verifies a user is able to join a project with a project code
Test Objective	FR8: Join.Project
Prerequisites	User is logged in

Test Case Id	20
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on the Join button on the Projects page	<input type="checkbox"/>
Expected Result: Join Project modal is displayed	
2. Enter a valid project code (found on project settings page -> Get Project ID) and click the Join button	
Expected Result: Successfully Joined Project toast message is displayed.	<input type="checkbox"/>
Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

<b>Priority</b>	Low ▾
<b>Description</b>	This test case verifies there is error checking on the fields of the Join Project modal
<b>Test Objective</b>	FR8: Join.Project
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	21
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
1. Click on the Join button on the Projects page	<input type="checkbox"/>
Expected Result: Join Project modal is displayed	
2. Click on the Project Code field and click off the Project Code field	
Expected Result: ‘Project code required’ error message is displayed.	<input type="checkbox"/>
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies there is error checking for an invalid project code
<b>Test Objective</b>	FR8: Join.Project
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	22
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
1. Click on the Join button on the Projects page	<input type="checkbox"/>
Expected Result: Join Project modal is displayed	
2. Enter 'x' in the Project Code field and click the Join button	
Expected Result: Joining Project Failed error toast message is displayed.	<input type="checkbox"/>
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

<b>Priority</b>	Low ▾
<b>Description</b>	This test case verifies users can cancel joining a project
<b>Test Objective</b>	FR8: Join.Project
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	23
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
1. Click on the Join button on the Projects page	<input type="checkbox"/>
Expected Result: Join Project modal is displayed	
2. Click on the Close button	
Expected Result: Join Project modal is closed.	<input type="checkbox"/>
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR9: Leave.Project

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies a project member is able to leave a project if they are not the project administrator
<b>Test Objective</b>	FR9: Leave.Project
<b>Prerequisites</b>	User is logged in User does not own the selected project

<b>Test Case Id</b>	24
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project not owned by the user</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed for the project	
<b>2. Click on the Settings button</b>	<input type="checkbox"/>
Expected Result: Settings page is displayed	
<b>3. Click on the Leave Project button</b>	<input type="checkbox"/>
Expected Result: Confirm Leave Project modal is displayed	
<b>5. Click on the Leave button</b>	<input type="checkbox"/>
Expected Result: Successfully Exited Project success toast message is displayed	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies a project member is not able to leave a project if they are the project administrator
<b>Test Objective</b>	FR9: Leave.Project
<b>Prerequisites</b>	User is logged in User owns the selected project

<b>Test Case Id</b>	25
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project owned by the user</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed for the project	
<b>2. Click on the Settings button</b>	<input type="checkbox"/>
Expected Result: Settings page is displayed. There is a Delete Project button in place of the Leave Project button	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

<b>Priority</b>	Medium ▾
<b>Description</b>	This test case verifies a project member is able to cancel leaving a project
<b>Test Objective</b>	FR9: Leave.Project
<b>Prerequisites</b>	User is logged in User does not own the selected project

<b>Test Case Id</b>	26
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project not owned by the user</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed for the project	
<b>2. Click on the Settings button</b>	<input type="checkbox"/>
Expected Result: Settings page is displayed	
<b>3. Click on the Leave Project button</b>	<input type="checkbox"/>
Expected Result: Confirm Leave Project modal is displayed	
<b>4. Click on the Cancel button</b>	<input type="checkbox"/>
Expected Result: Confirm Leave Project modal is closed	

<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR10: Add.Members.Project

Priority	High ▾
Description	This test case verifies a project administrator is able to add members to the project
Test Objective	FR10: Add.Members.Project
Prerequisites	User is logged in User owns the selected project

Test Case Id	27
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project the user owns	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
2. Click on the Settings button	<input type="checkbox"/>
Expected Result: Settings page is displayed	
3. Click on the Add Member button	<input type="checkbox"/>
Expected Result: Add Project Member modal is displayed	
4. Enter a the email of a user registered with the system who is not already a member on the project and click the Add button	<input type="checkbox"/>
Expected Result: Successfully Added Member To Project success toast message is displayed	
Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

<b>Priority</b>	Medium ▾
<b>Description</b>	This test case verifies there is error checking on Add Project Member field
<b>Test Objective</b>	FR10: Add.Members.Project
<b>Prerequisites</b>	User is logged in User owns the selected project

<b>Test Case Id</b>	28
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project the user owns</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Settings button</b>	<input type="checkbox"/>
Expected Result: Settings page is displayed	
<b>3. Click on the Add Member button</b>	<input type="checkbox"/>
Expected Result: Add Project Member modal is displayed	
<b>4. Click on the Email field and click off the email field</b>	<input type="checkbox"/>
Expected Result: ‘Email required’ error message is displayed	
<b>5. Enter ‘x’ in the Email field and click the Add button</b>	<input type="checkbox"/>
Expected Result: ‘Please include an ‘@’ in the email address. ‘x’ is missing an ‘@’.’ error message is displayed	

<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

<b>Priority</b>	Medium ▾
<b>Description</b>	This test case verifies there is error checking when trying to add a member with an email that does not exist in the system
<b>Test Objective</b>	FR10: Add.Members.Project
<b>Prerequisites</b>	User is logged in User owns the selected project

<b>Test Case Id</b>	29
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project the user owns</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Settings button</b>	<input type="checkbox"/>
Expected Result: Settings page is displayed	
<b>3. Click on the Add Member button</b>	<input type="checkbox"/>
Expected Result: Add Project Member modal is displayed	
<b>4. Enter the email of a user that does not exist in the system and click on the Add button</b>	<input type="checkbox"/>
Expected Result: Add Project Member Failed error toast message displayed	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies a project member cannot add other members if they are not the project administrator
<b>Test Objective</b>	FR10: Add.Members.Project
<b>Prerequisites</b>	User is logged in User does not own the selected project

<b>Test Case Id</b>	30
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project not owned by the user</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Settings button</b>	<input type="checkbox"/>
Expected Result: Settings page is displayed. There is no Project Member list displayed and no Add Member button	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR11: Delete.Member.Project

Priority	High
Description	This test case verifies a project administrator is able to remove members from the project
Test Objective	FR11: Delete.Member.Project
Prerequisites	User is logged in User owns the selected project Project has members other than the project administrator

Test Case Id	31
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project owned by the user	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
2. Click on the Settings button	<input type="checkbox"/>
Expected Result: Settings page is displayed	
3. Click on the Delete button beside a project member	<input type="checkbox"/>
Expected Result: Confirm Remove Member modal is displayed	
4. Click on the Remove button	<input type="checkbox"/>
Expected Result: Successfully Removed {Member Username} From Project success toast message is displayed	
Comment	
Execution status	Pass
Bug Severity	None

Bug Priority	None ▾
--------------	--------

<b>Priority</b>	Medium ▾
<b>Description</b>	This test case verifies a project administrator is able to cancel removing members from the project
<b>Test Objective</b>	FR11: Delete.Member.Project
<b>Prerequisites</b>	User is logged in User owns the selected project Project has members other than the project administrator

<b>Test Case Id</b>	32
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project owned by the user</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Settings button</b>	<input type="checkbox"/>
Expected Result: Settings page is displayed	
<b>3. Click on the Delete button beside a project member</b>	<input type="checkbox"/>
Expected Result: Confirm Remove Member modal is displayed	
<b>4. Click on the Cancel button</b>	<input type="checkbox"/>
Expected Result: Confirm Remove Member modal is closed	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies a project member cannot remove other members from the project if they are not the project administrator
<b>Test Objective</b>	FR11: Delete.Member.Project
<b>Prerequisites</b>	User is logged in User does not own the selected project Project has members other than the project administrator

<b>Test Case Id</b>	33
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project not owned by the user</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Settings button</b>	<input type="checkbox"/>
Expected Result: Settings page is displayed. Project Members list is not displayed and there is no delete member button.	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR12: View.Members.Project

Priority	High ▾
Description	This test case verifies project administrators can view a list of members on the project
Test Objective	FR12: View.Members.Project
Prerequisites	User is logged in User owns the selected project

Test Case Id	34
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project owned by the user	<input type="checkbox"/>
Expected Result: Kanban page displayed	
2. Click on the Settings button	<input type="checkbox"/>
Expected Result: Settings page is displayed. A list of project members is displayed on the settings page under the project name and description.	
Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies project members cannot view a list of members on the project
<b>Test Objective</b>	FR12: View.Members.Project
<b>Prerequisites</b>	User is logged in User does not own the selected project

<b>Test Case Id</b>	35
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
1. Click on a project not owned by the user	<input type="checkbox"/>
Expected Result: Kanban page displayed	
2. Click on the Settings button	
Expected Result: Settings page is displayed. The list of project members is not displayed	<input type="checkbox"/>
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR13: View.All.Projects

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies user can view a list of projects they own or are a member on
<b>Test Objective</b>	FR13: View.All.Projects
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	36
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
1. Navigate to the Projects page	<input type="checkbox"/>
Expected Result: List of all projects the user owns or is a member on is displayed	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR14: Add.Milestone

Priority	High ▾
Description	This test case verifies a user can add a milestone
Test Objective	FR14: Add.Milestone
Prerequisites	User is logged in

Test Case Id	37
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
2. Click on the Add button	<input type="checkbox"/>
Expected Result: Dropdown menu is displayed	
3. Click on Add Milestone	<input type="checkbox"/>
Expected Result: Create Milestone page is displayed	
4. Enter a Milestone Name, Description, modify the Due Date and click the Create Milestone button	<input type="checkbox"/>
Expected Result: Milestone Created success toast is displayed	

Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies there is error checking on fields on the Create Milestone page
<b>Test Objective</b>	FR14: Add.Milestone
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	38
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Add button</b>	<input type="checkbox"/>
Expected Result: Dropdown menu is displayed	
<b>3. Click on Add Milestone</b>	<input type="checkbox"/>
Expected Result: Create Milestone page is displayed	
<b>4. Click on the Milestone Name field and click off the Milestone Name field</b>	<input type="checkbox"/>
Expected Result: ‘Milestone name required’ error message is displayed	
<b>5. Click on the Description field and click off the Description field</b>	<input type="checkbox"/>
Expected Result: ‘Milestone description required’ error message is displayed	

<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

<b>Priority</b>	Medium ▾
<b>Description</b>	This test case verifies a user can cancel creating a milestone
<b>Test Objective</b>	FR14: Add.Milestone
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	39
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Add button</b>	<input type="checkbox"/>
Expected Result: Dropdown menu is displayed	
<b>3. Click on Add Milestone</b>	<input type="checkbox"/>
Expected Result: Create Milestone page is displayed	
<b>4. Click on the Cancel button</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR15: Edit.Milestone

Priority	High ▾
Description	This test case verifies a user can edit a milestone
Test Objective	FR15: Edit.Milestone
Prerequisites	User is logged in

Test Case Id	40
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
2. Click on the Edit button on a milestone	<input type="checkbox"/>
Expected Result: Edit Milestone modal is displayed	
3. Modify the Milestone Name, Description, and Due Date and Click the Update Milestone button	<input type="checkbox"/>
Expected Result: Edit Milestone modal is closed. Milestone Updated success modal is displayed. Updated Milestone is displayed on the Kanban board.	
Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies there is error checking on fields on the Edit Milestone page
<b>Test Objective</b>	FR15: Edit.Milestone
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	41
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Edit button on a milestone</b>	<input type="checkbox"/>
Expected Result: Edit Milestone modal is displayed	
<b>3. Remove Milestone Name and click off Milestone Name field</b>	<input type="checkbox"/>
Expected Result: ‘Milestone name required’ error message is displayed	
<b>4. Remove Description and click off the Description field</b>	<input type="checkbox"/>
Expected Result: ‘Milestone description required’ error message is displayed	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

<b>Priority</b>	Medium ▾
<b>Description</b>	This test case verifies a user can cancel editing a milestone
<b>Test Objective</b>	FR15: Edit.Milestone
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	42
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Edit button on a milestone</b>	<input type="checkbox"/>
Expected Result: Edit Milestone modal is displayed	
<b>3. Click on the Cancel Button</b>	<input type="checkbox"/>
Expected Result: Edit Milestone modal is closed. The milestone is not changed.	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR16: Delete.Milestone

Priority	High ▾
Description	This test case verifies users are able to delete milestones
Test Objective	FR16: Delete.Milestone
Prerequisites	User is logged in

Test Case Id	43
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
2. Click on the Delete button on a milestone	<input type="checkbox"/>
Expected Result: Delete Milestone modal is displayed	
3. Click on the Delete Milestone button	<input type="checkbox"/>
Expected Result: Milestone Deleted success toast message is displayed. The milestone is not displayed on the Kanban board.	

Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies users are able to cancel deleting a milestone
<b>Test Objective</b>	FR16: Delete.Milestone
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	44
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Delete button on a milestone</b>	<input type="checkbox"/>
Expected Result: Delete Milestone modal is displayed	
<b>3. Click on the Cancel button</b>	<input type="checkbox"/>
Expected Result: Delete Milestone modal is closed	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR17: Add.Task & FR18: Add.QA.Task

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies a user can add a task
<b>Test Objective</b>	FR17: Add.Task & FR18: Add.QA.Task
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	45
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Add button</b>	<input type="checkbox"/>
Expected Result: Dropdown menu is displayed	
<b>3. Click on Add Task</b>	<input type="checkbox"/>
Expected Result: Create Task and Create QA Task page is displayed	
<b>4. Enter the required values for the task and QA task</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed. Task Created success toast is displayed. Task is displayed in the To Do column.	

<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

<b>Priority</b>	High <input type="button" value="▼"/>
<b>Description</b>	This test case verifies there is error checking on fields on the Create Task / Create QA Task page
<b>Test Objective</b>	FR17: Add.Task & FR18: Add.QA.Task
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	46
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Add button</b>	<input type="checkbox"/>
Expected Result: Dropdown menu is displayed	
<b>3. Click on Add Task</b>	<input type="checkbox"/>
Expected Result: Create Milestone page is displayed	
<b>4. Click on the Task Name field and click off the Task Name field</b>	<input type="checkbox"/>
Expected Result: 'Task name required' error message is displayed	
<b>5. Click on the Task Description field and click off the Task Description field</b>	<input type="checkbox"/>
Expected Result: 'Task description required' error message is displayed	
<b>6. Click on the QA Task Name field and click off the QA Task Name field</b>	<input type="checkbox"/>
Expected Result: 'QA Task name required' error message is displayed	
<b>7. Click on the Task Description field and click off the Task Description field</b>	<input type="checkbox"/>
Expected Result: 'QA Task description required' error message is displayed	
<b>Comment</b>	

Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

<b>Priority</b>	Medium ▾
<b>Description</b>	This test case verifies a user can cancel creating a task
<b>Test Objective</b>	FR17: Add.Task & FR18: Add.QA.Task
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	47
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Add button</b>	<input type="checkbox"/>
Expected Result: Dropdown menu is displayed	
<b>3. Click on Add Task</b>	<input type="checkbox"/>
Expected Result: Create Task page is displayed	
<b>4. Click on the Cancel button</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR19: Filter.Task

Priority	High ▾
Description	This test case verifies users can filter tasks/QA task/milestones displayed on the Kanban board
Test Objective	FR19: Filter.Tasks
Prerequisites	User is logged in

Test Case Id	48
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
2. Click on the filter to display Tasks	<input type="checkbox"/>
Expected Result: Tasks are displayed on the Kanban board	
3. Click on the filter to display task by	<input type="checkbox"/>
Expected Result: Task should be displayed according to the filter	
Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

## FR20: Edit.Task

Priority	High ▾
Description	This test case verifies a user can edit a task & QA task
Test Objective	FR20: Edit.Task
Prerequisites	User is logged in

Test Case Id	49
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
2. Click on the Edit button on a task	<input type="checkbox"/>
Expected Result: Edit Task modal is displayed	
3. Modify the task and QA task fields and click the Update Task button	<input type="checkbox"/>
Expected Result: Edit Task modal is closed. Task Updated success modal is displayed. Updated Task and QA is displayed on the Kanban board.	

Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies there is error checking on fields on the Edit Task / Edit QA Task page
<b>Test Objective</b>	FR20: Edit.Task
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	50
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Edit button on a Task</b>	<input type="checkbox"/>
Expected Result: Edit Task modal is displayed	
<b>3. Remove Task Name and click off the Task Name field</b>	<input type="checkbox"/>
Expected Result: ‘Task name required’ error message is displayed	
<b>4. Remove the Task Description and click off the Task Description field</b>	<input type="checkbox"/>
Expected Result: ‘Task description required’ error message is displayed	
<b>8. Remove QA Task Name and click off the QA Task Name field</b>	<input type="checkbox"/>
Expected Result: ‘QA Task name required’ error message is displayed	
<b>9. Remove the Task Description and click off the Task Description field</b>	<input type="checkbox"/>
Expected Result: ‘QA Task description required’ error message is displayed	

<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾

Bug Priority	None ▾
--------------	--------

<b>Priority</b>	Medium ▾
<b>Description</b>	This test case verifies a user can cancel editing a task
<b>Test Objective</b>	FR20: Edit.Task
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	51
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Edit button on a task</b>	<input type="checkbox"/>
Expected Result: Edit Task modal is displayed	
<b>3. Click on the Cancel Button</b>	<input type="checkbox"/>
Expected Result: Edit Task modal is closed. The task and QA task is not changed.	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR21: Delete.Task

Priority	High ▾
Description	This test case verifies users are able to delete tasks
Test Objective	FR21: Delete.Task
Prerequisites	User is logged in

Test Case Id	52
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
2. Click on the Delete button on a task	<input type="checkbox"/>
Expected Result: Delete Task modal is displayed	
3. Click on the Delete Task button	<input type="checkbox"/>
Expected Result: Task Deleted success toast message is displayed. The task and QA task is not displayed on the Kanban board.	

Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies users are able to cancel deleting a task
<b>Test Objective</b>	FR21: Delete.Task
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	53
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on the Delete button on a task</b>	<input type="checkbox"/>
Expected Result: Delete Task modal is displayed	
<b>4. Click on the Cancel button</b>	<input type="checkbox"/>
Expected Result: Delete Task modal is closed	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR22: Move.Task

Priority	High ▾
Description	This test case verifies users are able to move tasks/QA tasks/milestones between columns on the Kanban board
Test Objective	FR22: Move.Task
Prerequisites	User is logged in

Test Case Id	54
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
2. Click-and-Drag a task from one column to another	<input type="checkbox"/>
Expected Result: Task is displayed in the column it was dragged to	
Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

## FR23: View.Board

Priority	High ▾
Description	This test case verifies users are able to view a Kanban board for a selected project
Test Objective	FR23: View.Board
Prerequisites	User is logged in Project has task/QA tasks/milestones

Test Case Id	55
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project	<input type="checkbox"/>
Expected Result: Kanban page is displayed. Tasks/QA Tasks/milestones	
Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

## FR24: View.Timeline

Priority	High ▾
Description	This test case verifies users are able to view a timeline of upcoming tasks
Test Objective	FR24: View.Timeline
Prerequisites	User is logged in Project has task/QA tasks/milestones

Test Case Id	56
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
2. Click on Timeline in the navigation bar	<input type="checkbox"/>
Expected Result: Timeline page is displayed. Tasks/QA tasks/milestones that are not yet completed are displayed from their creation date to their due date.	
Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

## FR25: Add.Sprint

Priority	High ▾
Description	This test case verifies users are able to add a sprint
Test Objective	FR25: Add.Sprint
Prerequisites	User is logged in

Test Case Id	57
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
2. Click on Sprints in the navigation bar	<input type="checkbox"/>
Expected Result: Sprints page is displayed	
3. Click on the Add button	<input type="checkbox"/>
Expected Result: Create Sprint page is displayed	
4. Enter Sprint Name, Description, modify the Start-End Date and click the Create button	<input type="checkbox"/>
Expected Result: Sprints page is displayed Sprint Successfully created toast message is displayed. Sprint is displayed in list - sorted according to start date.	

Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies users are able to add milestones/tasks/QA tasks to a sprint
<b>Test Objective</b>	FR25: Add.Sprint
<b>Prerequisites</b>	User is logged in Sprint is created Project has a task and milestone unassigned to any sprint

<b>Test Case Id</b>	58
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on Sprints in the navigation bar</b>	<input type="checkbox"/>
Expected Result: Sprints page is displayed	
<b>3. Click on Sprint dropdown to view the sprint details</b>	<input type="checkbox"/>
Expected Result: Sprint description and list of tasks/milestones/QA tasks in the sprint are displayed	
<b>4. Click the Add Milestone/Task dropdown</b>	<input type="checkbox"/>
Expected Result: List of milestones/tasks unassigned to a sprint are displayed	
<b>5. Click the Add Milestone/Task dropdown</b>	<input type="checkbox"/>
Expected Result: List of milestones/tasks unassigned to a sprint are displayed	
<b>6. Click on a Task</b>	<input type="checkbox"/>
Expected Result: Task and QA task is displayed in the list of tasks/milestones/QA tasks assigned to the sprint	
<b>7. Click on a Milestone</b>	<input type="checkbox"/>

Expected Result: Milestone is displayed in the list of tasks/milestones/QA tasks assigned to the sprint

<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies there is error checking on the Create Sprint page fields
<b>Test Objective</b>	FR25: Add.Sprint
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	59
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on Sprints in the navigation bar</b>	<input type="checkbox"/>
Expected Result: Sprints page is displayed	
<b>3. Click on the Add button</b>	<input type="checkbox"/>
Expected Result: Create Sprint page is displayed	
<b>4. Click on the Sprint Name field and click off the Sprint Name field</b>	<input type="checkbox"/>
Expected Result: ‘Sprint name required’ error message is displayed	
<b>5. Click on the Start Date - End Date field</b>	<input type="checkbox"/>
Expected Result: Dates before today's date should not be selectable	

<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

<b>Priority</b>	Medium ▾
<b>Description</b>	This test case verifies users can cancel creating a sprint
<b>Test Objective</b>	FR25: Add.Sprint
<b>Prerequisites</b>	User is logged in

<b>Test Case Id</b>	60
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on Sprints in the navigation bar</b>	<input type="checkbox"/>
Expected Result: Sprints page is displayed	
<b>3. Click on the Add button</b>	<input type="checkbox"/>
Expected Result: Create Sprint page is displayed	
<b>4. Click on the Cancel button</b>	<input type="checkbox"/>
Expected Result: Sprints page is displayed	
<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR26: Delete.Sprint

Priority	High ▾
Description	This test case verifies users are able to delete a sprint
Test Objective	FR26: Delete.Sprint
Prerequisites	User is logged in Sprint is created

Test Case Id	61
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
2. Click on Sprints in the navigation bar	<input type="checkbox"/>
Expected Result: Sprints page is displayed	
3. Click on a Sprint dropdown to view the sprint details	<input type="checkbox"/>
Expected Result: Sprint details are displayed	
4. Click on the Delete button	<input type="checkbox"/>
Expected Result: Confirm Sprint Deletion modal is displayed	
5. Click on the Delete button	<input type="checkbox"/>
Expected Result: Sprint Deleted success toast is displayed. Deleted sprint is not displayed in the sprint list.	
Comment	
Execution status	Pass ▾

Bug Severity	<input type="button" value="None ▾"/>
Bug Priority	<input type="button" value="None ▾"/>

<b>Priority</b>	Medium ▾
<b>Description</b>	This test case verifies users are able to cancel deleting a sprint
<b>Test Objective</b>	FR26: Delete.Sprint
<b>Prerequisites</b>	User is logged in Sprint is created

<b>Test Case Id</b>	62
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on Sprints in the navigation bar</b>	<input type="checkbox"/>
Expected Result: Sprints page is displayed	
<b>3. Click on a Sprint dropdown to view the sprint details</b>	<input type="checkbox"/>
Expected Result: Sprint details are displayed	
<b>4. Click on the Delete button</b>	<input type="checkbox"/>
Expected Result: Confirm Sprint Deletion modal is displayed	
<b>5. Click on the Cancel button</b>	<input type="checkbox"/>
Expected Result: Sprint page is displayed. Sprint is still displayed in list	

<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR27: Edit Sprint

Priority	High ▾
Description	This test case verifies users are able to edit a sprint
Test Objective	FR27: Edit.Sprint
Prerequisites	User is logged in Sprint is created

Test Case Id	63
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
2. Click on Sprints in the navigation bar	<input type="checkbox"/>
Expected Result: Sprints page is displayed	
3. Click on a Sprint dropdown to view the sprint details	<input type="checkbox"/>
Expected Result: Sprint details are displayed	
4. Click on the Edit button	<input type="checkbox"/>
Expected Result: Edit Sprint page is displayed	
5. Modify the Sprint Name, Description, and Start Date - End Date and click the Save button	<input type="checkbox"/>
Expected Result: Sprint Successfully Updated success toast is displayed. Updated sprint is displayed in the sprint list.	
Comment	
Execution status	Pass ▾

Bug Severity	<input type="button" value="None ▾"/>
Bug Priority	<input type="button" value="None ▾"/>

<b>Priority</b>	High <input type="button" value="▼"/>
<b>Description</b>	This test case verifies users are able to remove milestones/tasks/QA tasks from a sprint
<b>Test Objective</b>	FR27: Edit.Sprint
<b>Prerequisites</b>	User is logged in Sprint is created Project has a tasks and milestone assigned to a sprint

<b>Test Case Id</b>	64
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on Sprints in the navigation bar</b>	<input type="checkbox"/>
Expected Result: Sprints page is displayed	
<b>3. Click on Sprint dropdown to view the sprint details</b>	<input type="checkbox"/>
Expected Result: Sprint details are displayed	
<b>4. Click the Delete button beside a Task</b>	<input type="checkbox"/>
Expected Result: Task and associated QA task are not displayed in the list of tasks/QA tasks/milestones in the sprint	
<b>5. Click the Delete button beside a Milestone</b>	<input type="checkbox"/>
Expected Result: Milestone is not displayed in the list of tasks/QA tasks/milestones in the sprint	
<b>Comment</b>	
<b>Execution status</b>	Pass <input type="button" value="▼"/>

Bug Severity	<input type="button" value="None ▾"/>
Bug Priority	<input type="button" value="None ▾"/>

<b>Priority</b>	High ▾
<b>Description</b>	This test case verifies there is error checking on the Edit Sprint page fields
<b>Test Objective</b>	FR27: Edit.Sprint
<b>Prerequisites</b>	User is logged in Sprint is created

<b>Test Case Id</b>	65
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on Sprints in the navigation bar</b>	<input type="checkbox"/>
Expected Result: Sprints page is displayed	
<b>3. Click on a Sprint dropdown to view the sprint details</b>	<input type="checkbox"/>
Expected Result: Sprint details are displayed	
<b>4. Click on the Edit button</b>	<input type="checkbox"/>
Expected Result: Edit Sprint page is displayed	
<b>5. Remove the Sprint Name and click off the Sprint Name field</b>	<input type="checkbox"/>
Expected Result: ‘Sprint name required’ error message is displayed	
<b>6. Click on the Start Date - End Date field</b>	<input type="checkbox"/>
Expected Result: Dates before the current sprint start Date should not be selectable	

<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾

Bug Priority	None ▾
--------------	--------

<b>Priority</b>	Medium ▾
<b>Description</b>	This test case verifies users are able to cancel editing a sprint
<b>Test Objective</b>	FR27: Edit.Sprint
<b>Prerequisites</b>	User is logged in Sprint is created

<b>Test Case Id</b>	66
<b>Test Case Reviewer</b>	Alex Liu
<b>Test Case Version</b>	1.0
<b>Test Execution Date</b>	Apr 5, 2024

<b>Test Steps</b>	
<b>1. Click on a project</b>	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
<b>2. Click on Sprints in the navigation bar</b>	<input type="checkbox"/>
Expected Result: Sprints page is displayed	
<b>3. Click on a Sprint dropdown to view the sprint details</b>	<input type="checkbox"/>
Expected Result: Sprint details are displayed	
<b>4. Click on the Edit button</b>	<input type="checkbox"/>
Expected Result: Edit Sprint page is displayed	
<b>5. Click on the Cancel button</b>	<input type="checkbox"/>
Expected Result: Sprints page is displayed. Sprint is unchanged	

<b>Comment</b>	
<b>Execution status</b>	Pass ▾
<b>Bug Severity</b>	None ▾
<b>Bug Priority</b>	None ▾

## FR28: View.Sprints

Priority	High ▾
Description	This test case verifies users are able to view sprints
Test Objective	FR28: View.Sprints
Prerequisites	User is logged in At least two sprints are created for the selected project

Test Case Id	67
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
2. Click on Sprints in the navigation bar	<input type="checkbox"/>
Expected Result: Sprints page is displayed. A list of sprints is displayed	
3. Click on a Sprint dropdown to view the sprint details	<input type="checkbox"/>
Expected Result: Sprint details are displayed	
Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

## FR29: View.Project.Dependencies.Alpha

Priority	High ▾
Description	This test case verifies users are able to view project dependencies and the alpha completeness
Test Objective	FR29: View.Project.Dependencies.Alpha
Prerequisites	User is logged in Selected project has multiple tasks/QA tasks/milestones

Test Case Id	68
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
2. Click on Dependencies in the navigation bar	<input type="checkbox"/>
Expected Result: Dependency Analysis page is displayed. By default both completed and not completed tasks/milestones are displayed. The Alpha Completeness percentage is displayed (tasks are considered Alpha complete when their status is 'completed'. The associated QA task may or may not be complete)	
Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

## FR30: View.Project.Dependencies.QA

Priority	High ▾
Description	This test case verifies users are able to view project dependencies and the QA completeness
Test Objective	FR30: View.Project.Dependencies.QA
Prerequisites	User is logged in Selected project has multiple tasks/QA tasks/milestones

Test Case Id	69
Test Case Reviewer	Alex Liu
Test Case Version	1.0
Test Execution Date	Apr 5, 2024

Test Steps	
1. Click on a project	<input type="checkbox"/>
Expected Result: Kanban page is displayed	
2. Click on Dependencies in the navigation bar	<input type="checkbox"/>
Expected Result: Dependency Analysis page is displayed	
3. Click on the Toggle QA Tasks	<input type="checkbox"/>
Expected Result: Completed and not completed tasks/milestones and QA tasks are displayed. The QA Completeness percentage is displayed (both the task and QA task must be completed to be considered QA complete).	
Comment	
Execution status	Pass ▾
Bug Severity	None ▾
Bug Priority	None ▾

## Backend Tests

Name	Last Modified	Reason for Changes	Version
Elena Arnold Youssef Ismail Alex Liu	Apr 5, 2024	Initial version	1.0

Total Test Cases	Last Executed	Passed Test Cases	Pass %
105	Apr 5, 2024	105	100%

### Coverage Report:

Name	Stmts	Miss	Cover
api/__init__.py	0	0	100%
api/database.py	79	0	100%
api/main.py	19	0	100%
api/migrations/001.py	6	6	0%
api/migrations/002.py	7	7	0%
api/migrations/__init__.py	0	0	100%
api/routers/__init__.py	12	0	100%
api/routers/milestones.py	48	0	100%
api/routers/projects.py	104	4	96%
api/routers/sprints.py	48	0	100%
api/routers/tasks.py	57	0	100%
api/routers/users.py	55	0	100%
api/schemas.py	120	0	100%
api/tests/__init__.py	0	0	100%
api/tests/test_database.py	13	0	100%
api/tests/test_milestones.py	166	0	100%
api/tests/test_projects.py	317	0	100%
api/tests/test_sprints.py	135	0	100%
api/tests/test_tasks.py	238	0	100%
api/tests/test_users.py	96	0	100%
api/tests/util.py	23	0	100%
TOTAL	1543	17	99%

Test Case ID	Test Description	Test Steps	Expected Results
DB-001	Verify the <code>getDb</code> function returns a database object.	1. Call the <code>getDb</code> function.	1. A database object is returned.
DB-002	Verify <code>toObjectId</code> correctly handles a valid object ID.	1. Provide a valid object ID string ("5f6e7d5f7e3f5e7d5f7e3f5e") to the <code>toObjectId</code> function.	1. A valid <code>ObjectId</code> is returned.
DB-003	Verify <code>toObjectId</code> raises an <code>HTTPException</code> for an invalid ID.	1. Provide an invalid object ID string ("123") to the <code>toObjectId</code> function.	1. An <code>HTTPException</code> is raised.
MIL-001	Verify successful milestone creation	1. Create a user. 2. Create a project with the user. 3. Call the <code>createMilestone</code> endpoint with valid data.	1. Status code 200 is returned. 2. Response contains newly created milestone data.
MIL-002	Verify creation fails if the project does not exist	1. Create a user. 2. Call the <code>createMilestone</code> endpoint with valid data and a non-existent project ID.	1. Status code 404 is returned.
MIL-003	Verify creation fails if the user doesn't have access to the project	1. Create two users (user and otherUser). 2. Create a project with user. 3. Call the <code>createMilestone</code> endpoint with otherUser and the project ID.	1. Status code 403 is returned.
MIL-004	Verify creation fails with a database error	1. Mock <code>insert_one</code> to return <code>acknowledged=False</code> . 2. Create a user, project. 3. Call <code>createMilestone</code> .	1. Status code 500 is returned.
MIL-005	Verify successful milestone retrieval	1. Create user, project, and milestone. 2. Call <code>getMilestone</code> endpoint with the milestone ID.	1. Status code 200 is returned. 2. Response contains correct milestone data.
MIL-006	Verify retrieval fails if the milestone does not exist	1. Create a user. 2. Call the <code>getMilestone</code> endpoint with a non-existent milestone ID	1. Status code 404 is returned.
MIL-007	Verify retrieval fails if the user doesn't have access to the milestone	1. Create users (user, otherUser), project, and milestone (with user). 2. Call <code>getMilestone</code> with otherUser and the milestone ID.	1. Status code 403 is returned.
MIL-008	Verify successful milestone update	1. Create user, project, and milestone. 2. Call <code>updateMilestone</code> endpoint with new data and milestone ID.	1. Status code 200 is returned. 2. Response contains updated milestone data.
MIL-009	Verify update fails if the milestone does not exist	1. Mock <code>find_one</code> to return <code>None</code> . 2. Create user, project, and call <code>updateMilestone</code> with non-existent milestone ID.	1. Status code 404 is returned.
MIL-010	Verify update fails with database error	1. Mock <code>find_one_and_update</code> to return <code>None</code> . 2. Create user, project, milestone. 3. Call <code>updateMilestone</code> .	1. Status code 500 is returned.

MIL-011	Verify update fails if the user doesn't have access to the milestone	1. Create users, project, milestone. 2. Call updateMilestone with otherUser and milestone ID.	1. Status code 403 is returned.
MIL-012	Verify successful milestone deletion	1. Create user, project, milestone. 2. Call deleteMilestone. 3. Attempt to retrieve the milestone.	1. Status code 200 on delete. 2. Status code 404 when attempting to retrieve.
MIL-013	Verify deletion fails if the milestone does not exist	1. Mock find_one to return None. 2. Create user, project. 3. Call deleteMilestone with non-existent ID.	1. Status code 404 is returned.
MIL-014	Verify deletion fails with database error (delete_one)	1. Mock delete_one (return .deleted_count = 0). 2. Create user, project, milestone. 3. Call deleteMilestone.	1. Status code 500 is returned.
MIL-015	Verify deletion fails with database error (update_many) on milestones	1. Mock update_many to fail on milestones. 2. Create user, project, milestone. 3. Call deleteMilestone.	1. Status code 500 is returned.
MIL-016	Verify deletion fails with database error (update_many) on tasks	1. Mock update_many to fail on tasks. 2. Create user, project, milestone. 3. Call deleteMilestone.	1. Status code 500 is returned.
PROJ-001	Verify successful project creation	1. Create a user. 2. Call createProject endpoint with valid data, including project name and description.	1. Status code 200 is returned. 2. Response contains newly created project data.
PROJ-002	Verify creation fails with database error (insert_one)	1. Mock insert_one to return acknowledged=False. 2. Create a user and project data. 3. Call createProject endpoint.	1. Status code 500 is returned.
PROJ-003	Verify creation fails if the user cannot be found (find_one_and_update)	1. Mock find_one_and_update to return None. 2. Create user data. 3. Call createProject.	1. Status code 500 is returned.
PROJ-004	Verify project retrieval (including milestones, tasks, and sprints)	1. Create a user, project, milestone, task, and sprint (associated with the project). 2. Call getProject endpoint with the project's ID.	1. Status code 200 is returned. 2. Response contains correct project, milestone, task, and sprint data.
PROJ-005	Verify retrieval fails if the project does not exist	1. Create user. 2. Call getProject endpoint with a non-existent project ID.	1. Status code 404 is returned.
PROJ-006	Verify retrieval fails if the user doesn't have access to the project	1. Create users (user and otherUser), project, with user. 2. Call getProject with otherUser and project ID.	1. Status code 403 is returned.
PROJ-007	Verify successful project deletion	1. Create user, project, related milestone, task, and sprint. 2. Call deleteProject. 3. Attempt to retrieve the project, milestone, task and sprint.	1. Status code 200 on delete. 2. Status code 404 when attempting to retrieve entities.

PROJ-008	Verify deletion fails if the project does not exist	1. Mock find_one to return None. 2. Create a user. 3. Call deleteProject with non-existent ID.	1. Status code 404 is returned.
PROJ-009	Verify deletion fails with database error on projects (delete_one)	1. Mock delete_one in projects to fail. 2. Create user, project. 3. Call deleteProject.	1. Status code 500 is returned.
PROJ-010	Verify deletion fails with database error on milestones (delete_many)	1. Mock delete_many in milestones to fail. 2. Create user, project. 3. Call deleteProject.	1. Status code 500 is returned.
PROJ-011	Verify deletion fails with database error on tasks (delete_many)	1. Mock delete_many in tasks to fail. 2. Create user, project. 3. Call deleteProject.	1. Status code 500 is returned.
PROJ-012	Verify deletion fails with database error on sprints (delete_many)	1. Mock delete_many in sprints to fail. 2. Create user, project. 3. Call deleteProject.	1. Status code 500 is returned.
PROJ-013	Verify deletion fails with database error on users (find_one_and_update)	1. Mock find_one_and_update on users to fail. 2. Create user and project. 3. Call deleteProject.	1. Status code 500 is returned.
PROJ-014	Verify deletion fails with database error on users (update_many)	1. Mock update_many on users to fail. 2. Create user and project. 3. Call deleteProject.	1. Status code 500 is returned.
PROJ-015	Verify project update	1. Create user, project. 2. Call updateProject with new name/description and project ID.	1. Status code 200 is returned. 2. Response contains updated project data.
PROJ-016	Verify update fails if the project does not exist	1. Mock find_one_and_update to return None. 2. Create user, project data. 3. Call 'updateProject' with non-existent project ID.	1. Status code 404 is returned.
PROJ-017	Verify update fails with database error (find_one_and_update)	1. Mock find_one_and_update to return None. 2. Create project, user, and update data. 3. Call updateProject.	1. Status code 500 is returned.
PROJ-018	Verify user can join a project	1. Create users (user and user2), project (with user). 2. Call joinProject endpoint (user2).	1. Status code 200. 2. User data (user2) reflects joined project.
PROJ-019	Verify join fails with a non-existent project	1. Create user. 2. Call joinProject with non-existent project ID.	1. Status code 404 is returned.
PROJ-020	Verify joining fails with database error (find_one_and_update)	1. Mock find_one_and_update to fail. 2. Create user, project.	1. Status code 500 is returned.
SPRT-001	Verify successful sprint creation	1. Create user, project. 2. Call createSprint endpoint with valid data, including dates and project ID.	1. Status code 200 is returned. 2. Response contains newly created sprint data.
SPRT-002	Verify creation fails if the project	1. Create user. 2. Call createSprint	1. Status code 404 is returned.

	does not exist	endpoint with valid data and non-existent project ID.	
SPRT-003	Verify creation fails with unauthorized user	1. Create users (user and otherUser), project (with user). 2. Call createSprint with otherUser and project ID.	1. Status code 403 is returned.
SPRT-004	Verify creation fails with database error (insert_one)	1. Mock insert_one to return acknowledged=False. 2. Create a user, project, and sprint data. 3. Call createSprint.	1. Status code 500 is returned.
SPRT-005	Verify sprint retrieval	1. Create user, project, and sprint. 2. Call getSprint endpoint with the sprint ID.	1. Status code 200 is returned. 2. Response contains correct sprint data.
SPRT-006	Verify retrieval fails with unauthorized user	1. Create users (user, otherUser), project, sprint (with user). 2. Call getSprint with otherUser and sprint ID.	1. Status code 403 is returned.
SPRT-007	Verify retrieval fails if the sprint does not exist	1. Create user, project. 2. Call getSprint with non-existent sprint ID.	1. Status code 404 is returned.
SPRT-008	Verify successful sprint update	1. Create user, project, and sprint. 2. Call updateSprint endpoint with new data and sprint ID.	1. Status code 200 is returned. 2. Response contains updated sprint data.
SPRT-009	Verify update fails if the sprint does not exist	1. Mock find_one_and_update to return None. 2. Create a user, project and sprint data. 3. Call updateSprint.	1. Status code 404 is returned.
SPRT-010	Verify update fails with unauthorized user	1. Create users (user, otherUser), project, and sprint (with user). 2. Call updateSprint with otherUser and sprint ID.	1. Status code 403 is returned.
SPRT-011	Verify update fails with database error (find_one_and_update)	1. Mock find_one_and_update to return None. 2. Create a user, project, sprint, and updates. 3. Call updateSprint.	1. Status code 500 is returned.
SPRT-012	Verify successful sprint deletion	1. Create user, project, sprint. 2. Call deleteSprint. 3. Attempt to retrieve sprint.	1. Status code 200 on delete. 2. Status code 404 when attempting to retrieve.
SPRT-013	Verify deletion fails if the sprint does not exist	1. Mock find_one to return None. 2. Create a user and project. 3. Call deleteSprint with non-existent ID.	1. Status code 404 is returned.
SPRT-014	Verify deletion fails with unauthorized user	1. Create users (user, otherUser), project, and sprint (with user). 2. Call deleteSprint with otherUser and sprint ID.	1. Status code 403 is returned.
SPRT-015	Verify deletion fails with database error (delete_one)	1. Mock delete_one to fail. 2. Create user, project, and sprint. 3. Call	1. Status code 500 is returned.

		deleteSprint.	
TASK-001	Verify successful task creation	1. Create user, project, and milestone. 2. Call createTask endpoint with valid data, including project and milestone IDs, and the task payload (i.e., self.testTask).	1. Status code 200 is returned. 2. Response contains newly created task data.
TASK-002	Verify creation fails with non-existent project	1. Create user and milestone. 2. Call createTask with valid data, non-existent project ID, and milestone ID.	1. Status code 404 is returned.
TASK-003	Verify creation fails with non-existent milestone	1. Create user and project. 2. Call createTask with valid data, project ID, and non-existent milestone ID.	1. Status code 404 is returned.
TASK-004	Verify creation fails with mismatched project and milestone	1. Create users, two projects (project1, project2), and a milestone (in project1). 2. Call createTask with project2 ID, milestone ID, and valid task data.	1. Status code 400 is returned.
TASK-005	Verify creation fails with unauthorized user	1. Create users (user and anotherUser), project, and milestone (with user). 2. Call createTask with anotherUser, project ID and milestone ID.	1. Status code 403 is returned.
TASK-006	Verify creation fails with database error (insert_one)	1. Mock insert_one to return acknowledged=False. 2. Create user, project, milestone, and task data. 3. Call createTask.	1. Status code 500 is returned.
TASK-007	Verify task retrieval	1. Create user, project, milestone, and task. 2. Call getTask endpoint with the task ID.	1. Status code 200 is returned. 2. Response contains correct task data.
TASK-008	Verify retrieval fails with unauthorized user	1. Create users (user and anotherUser), project, milestone, and task (with user). 2. Call getTask with anotherUser and task ID.	1. Status code 403 is returned.
TASK-009	Verify successful task update	1. Create user, project, milestone, and task. 2. Call updateTask endpoint with new data and task ID, including any updates to qaTask.	1. Status code 200 is returned. 2. Response contains updated task data.
TASK-010	Verify task update with project and milestone change	1. Create two projects (project1, project2), two milestones (milestone1 in project1, milestone2 in project2), and a task in project1 associated with milestone1. 2. Call updateTask with the task ID, new project ID (project2), and new milestone ID (milestone2).	1. Status code 200 is returned. 2. Updated task has the correct project and milestone IDs.

TASK-011	Verify update fails when project does not exist	1. Create user, project, milestone, and task. 2. Call updateTask with a non-existent project ID.	1. Status code 404 is returned.
TASK-012	Verify update fails with non-existent milestone	1. Create user, project, milestone, and task. 2. Call updateTask with a non-existent milestone ID.	1. Status code 404 is returned.
TASK-013	Verify update fails with non-existent task	1. Mock find_one_and_update to return None. 2. Create user, project, milestone and update data. 3. Call updateTask with non-existent task ID.	1. Status code 404 is returned.
TASK-014	Verify update fails with unauthorized user	1. Create users (user and anotherUser), project, milestone, and task (with user). 2. Call updateTask with anotherUser and task ID.	1. Status code 403 is returned.
TASK-015	Verify update fails with database error (find_one_and_update)	1. Mock find_one_and_update to return None. 2. Create user, project, milestone, task, and update data. 3. Call updateTask.	1. Status code 500 is returned.
TASK-016	Verify task deletion	1. Create a user, project, milestone, and task. 2. Call deleteTask endpoint with the task ID.	1. Status code 200 on delete. 2. Status code 404 when attempting to retrieve the task.
TASK-017	Verify task deletion with dependent tasks	1. Create a user, project, milestone, and two tasks (task1, task2) where task2 has dependency on task1. 2. Call deleteTask with task1 ID.	1. Status code 200. 2. Verify task1 is deleted. 3. Verify task2 has empty dependencies.
TASK-018	Verify deletion fails with non-existent task	1. Create a user, project, and milestone. 2. Call deleteTask endpoint with a non-existent task ID.	1. Status code 404 is returned.
TASK-019	Verify deletion fails with unauthorized user	1. Create users (user, anotherUser), project, milestone, and task (created by user). 2. Call deleteTask with anotherUser and task ID.	1. Status code 403 is returned.
TASK-020	Verify deletion fails with database error (delete_one)	1. Mock delete_one to fail. 2. Create a user, project, milestone, and task. 3. Call deleteTask.	1. Status code 500 is returned.
TASK-021	Verify deletion failure updates milestones (update_many)	1. Mock update_many (in milestones collection) to fail. 2. Create a user, project, milestone, and task. 3. Call deleteTask.	1. Status code 500
USER-001	Verify successful user registration	1. Call registerUser endpoint with valid user data (username, password, email).	1. Status code 201 is returned. 2. Response contains user data (excluding raw password).
USER-002	Verify registration fails with invalid	1. Call registerUser with invalid data	1. Status code 422 is returned. 2.

	data	(e.g., missing email, short password).	Response contains error details.
USER-003	Verify registration fails if user already exists (username)	1. Register a user. 2. Call registerUser again with the same username.	1. Status code 400 is returned. 2. Response contains an appropriate error message.
USER-004	Verify registration fails if user already exists (email)	1. Register a user. 2. Call registerUser again with a different username but the same email.	1. Status code 400 is returned. 2. Response contains an appropriate error message.
USER-005	Verify registration fails with database error (insert_one)	1. Mock insert_one to return acknowledged=False. 2 . Call registerUser with valid data.	1. Status code 500 is returned.
USER-006	Verify registration fails with token generation error (find_one_and_update)	1. Mock find_one_and_update to return None. 2. Call registerUser with valid data.	1. Status code 500 is returned.
USER-007	Verify successful login	1. Register a user. 2. Call loginUser with the correct username and password.	1. Status code 200 is returned. 2. Response contains a valid access token.
USER-008	Verify login fails with invalid credentials (incorrect password)	1. Register a user. 2. Call loginUser with the correct username and an incorrect password.	1. Status code 401 is returned.
USER-009	Verify login fails with invalid credentials (non-existent user)	1. Call loginUser with a non-existent username and any password.	1. Status code 401 is returned.
USER-010	Verify getting current user (valid authorization)	1. Register a user. 2. Get access token (call loginUser). 3. Call /users/me with the token in the "Authorization: Bearer {token}" header.	1. Status code 200 is returned. 2. Response contains the correct user data.
USER-011	Verify getting current user fails with invalid token	1. Call /users/me with an invalid token in the "Authorization: Bearer {token}" header.	1. Status code 401 is returned.
USER-012	Verify current user not found (find_one failure)	1. Register a user. 2. Get a valid access token. 3. Mock find_one to return None. 4. Call /users/me with a valid token.	1. Status code 404 is returned.
USER-013	Verify successful password change	1. Register a user. 2. Get a valid access token. 3. Call /users/password/reset with a valid token and a new password.	1. Status code 200 is returned. 2. New password can be used to login successfully.
USER-014	Verify password change fails on database error (find_one_and_update)	1. Register a user. 2. Get a valid access token. 3. Mock find_one_and_update to return None. 4. Call /users/password/reset.	1. Status code 500 is returned.