Test Plan

for

Jira Project Management Tool

Prepared by

Mohammed Tanvir Hassan

22 May, 2025

Table of Contents

1. Project Overview

This test plan outlines the testing strategy for the Jira project management tool. The project involves analyzing and testing the core functionalities of Jira, using a combination of manual and automated QA testing. This tool is widely used for issue tracking and project management. The primary goal is to ensure that the selected key features of Jira function as expected, are user-friendly, and meet the acceptance criteria defined in the Project Requirement Document. The PRD identifies 10 key features of Jira that will be the focus of this testing cycle. These features are crucial for users to manage projects, track tasks, and collaborate effectively. This testing effort is part of the Deepchain Labs Internship Technical Test.

2. Testing Scope and Objectives

2.1 In Scope

The following features and functionalities of the Jira Tool are in scope for this test plan based on the PRD:

2.1.1. Login: Verifying that the user can sign in to the tool with valid credentials.
2.1.2. Issue Creation: Verifying the ability to create new issues with all relevant fields (summary, type, status, description, assignee, labels, etc.).
2.1.3. Task Creation: Verifying the creation of standard "Task" type issues.
2.1.4. Sub-task Creation: Verifying the creation and linking of sub-tasks to the parent issue.
2.1.5. Sprint Creation: Verifying the ability to create and define Sprints with goals and duration.
2.1.6. Adding Comment: Verifying the functionality of adding, editing, and viewing comments on issues.
2.1.7. Changing Status: Verifying the transition of issue statuses through the defined workflow (e.g., To Do → In Progress → In QA → Done)
2.1.8. Search Issue: Verifying the ability to search for issues using keywords, titles, and filters.
2.1.9. Assigning Ticket to User: Verifying the assignment and unassignment of issues to project members.
2.1.10. Adding Labels: Verifying the functionality of adding and managing labels for issues.

2.2 Out of Scope

The following aspects are considered out of scope for this testing cycle:

2.2.1. Performance Testing such as Load, Stress, Volume.
2.2.2. Security Testing
2.2.3. Testing of third-party integrations not mentioned in the PRD.
2.2.4. Database Integrity Testing.
2.2.5. API Level Testing.

2.3 Objective

The following aspects are considered objective for this testing cycle:
2.3.1. To verify that all in-scope Jira features function according to the specifications in the PRD.
2.3.2. To identify and report defects in the functionality, usability, and user interface of the selected Jira features.
2.3.3. To ensure that the acceptance criteria for each of the 10 key features are met.
2.3.4. To validate the usability of the core features from an end-user perspective.

3. Assumptions and Risks

3.1 Assumptions

3.1.1. The Jira tool will be available and stable during the testing period.
3.1**.2.** The tester has valid Jira access and permission to perform tasks.
3.1.3. The Project Requirement Document (PRD) is complete.
3.1**.4.** The necessary hardware and software for tests will be accessible.
3.1.5. Test data required for executing test cases will be available or can be created.

3.2 Risks

3.2.1. Unavailability of Jira Tool during testing.
3.2.2. Changes in product requirement documentation.
3.2.3. Insufficient test data.
3.2.4. Potential UI changes during testing due to live updates from Atlassian.
3.2.5. Automation failures if dynamic selectors change.

4. Types of Testing

The following types of testing will be performed:
4.1.1. Functional Testing: To verify that each function of the tool behaves as specified in the PRD and meets the acceptance criteria. This will be the primary focus for the 10 core features.
4.1.2. Usability Testing: To verify the ease of use and overall user experience of the Jira features.
4.1.3. Positive Testing: To verify that the system works as expected with valid inputs.
4.1.4. Negative Testing: To verify that the system handles invalid inputs and errors.
4.1.5. Smoke Testing: Initial tests on the build to check if the critical functionalities are working correctly before proceeding with more detailed testing.

5. Test Approach and Methodology

5.1 Test Approach

A hybrid approach will be adopted, combining manual testing for exploratory and usability aspects, and automated testing for the login feature only.

5.1.1. Manual Testing:

5.1.1.1. Exploratory testing to discover defects not covered by scripted test cases.
5.1.1.2. Usability testing to evaluate user-friendliness.
5.1.1.3. Executing test cases that are complex to automate.

5.1.2. Automated Testing:

5.1.2.1. A basic Selenium script in JavaScript will be developed to check login page email field.

5.2 Test Methodology

The following phases will be in testing:

5.2.1. Test Planning: Defining the scope, objectives, resources and schedule.
5.2.2. Test Case Design: Creating detailed test cases based on the PRD and selected features. Each feature will have 2-3 test cases covering positive and negative scenarios.
5.2.3. Test Environment Setup: Ensuring the test environment (browser, Jira) is ready.
5.2.4. Test Execution: Executing the designed test cases both manual and automated.
5.2.5. Test Reporting: Summarizing test execution progress, defects found and overall quality assessment.

6. Test Environment

Testing will be conducted in the following environment:

6.1.1. Platform: Web Application (Jira)
6.1.2. Operating System: Windows 11
6.1.3. Browsers: Google Chrome (Version 136+), Mozilla Firefox (Version 138+)
6.1.4. Internet Connectivity: Stable.
6.1.5. Test Data: Pre-created projects, users and sprints.

7. Test Entry and Exit Criteria

7.1 Entry Criteria

Testing activities for specific features will begin when the following conditions are met:

7.1.1. Jira login credentials are available.

7.1.2. The test environment setup is ready.

7.1.3. PRD and Test cases reviewed and approved.

7.2 Exit Criteria

Testing activities will be considered complete when the following conditions are met:

7.1.1. All test cases were executed.

7.1.2. All critical bugs are documented.

7.1.3. Automation script successfully runs and passes.

8. Roles and Responsibilities

Name	Role	Responsibility
Mohammed Tanvir Hassan	QA	- Develop Product Requirement Document. - Develop Test Plan. - Develop Test Cases. - Setup Environment. - Develop automation script. - Execute Test Cases for both manual and automated.

9. Tools to be Used

9.1 Manual Testing Tools

9.1.1. Web Browser: Google Chrome, Mozilla Firefox

9.2 Automation Testing Tools

9.2.1. Selenium WebDriver: For automating test scenarios.

9.2.2. Node.js: Runtime environment for JavaScript.

9.2.3. Mocha & Chai: For structuring and asserting automation tests.

9.3 Documentation

9.3.1. Microsoft Word: For creating PRD, Test Plan, Test Cases. Output to PDF as required**.**

10. Test Schedule

10.1 Estimated Effort:

Task	Estimated Effort (Hours)	Description
PRD Development	6-8	Drafting features, acceptance criteria, and goals
Test Plan	5-6	Structuring scope, risks, and methodology
Test Case Design	8-10	Writing 2–3 test cases per feature (10 features)
Automation Script Development	2-3	Writing a Selenium script for login page
Manual Testing Execution	10-12	Running test cases for all 10 features
Automation Execution	1-2	Running and debugging the Selenium script
Reporting & Documentation	3-4	Finalizing PDFs, README, and zip file
Total	35-45	

10.2 Timeline (Gantt Chart)

Tasks	21/05/2025	22/05/2025	23/05/2025	24/05/2025	25/05/2025	26/05/2025
1. PRD Development	■	■				
2. Test Plan Creation	■	■				
3. Test Case Design	■	■	■			
4. Automation Script Development			■	■		
5. Manual Testing Execution				■	■	
6. Automation Execution					■	
7. Reporting & Documentation						■