

Software Test Plan

For



Prepared By

Abdullah Al Faisal

Version 1.0

Product Name	Product Version	Test Plan Version
Jira	Jira Cloud	1.0

Revision History

Revision	Updated By	Comments	Date
01	Abdullah Al Faisal	Initial Draft	29/05/2025

Contents

1. PROJECT OVERVIEW.....	1
2. TESTING SCOPE AND OBJECTIVES	1
2.1 Scope:	1
2.2 Objectives:.....	2
3. ASSUMPTIONS & RISKS	2
3.1 Assumptions:	2
3.2 Risk:	3
4. TYPES OF TESTING	3
5. TEST APPROACH & METHODOLOGY	4
6. TEST ENVIRONMENT.....	4
7. TEST ENTRY/EXIT CRITERIA	5
7.1 Entry Criteria:.....	5
7.2 Exit Criteria:	5
8. ROLES AND RESPONSIBILITIES	5
9. TOOLS TO BE USED	6
10. TEST SCHEDULE	7

1. PROJECT OVERVIEW

The purpose of this quality assurance project is to verify the features of Jira. Jira is a project management and issue tracking tool that is widely used by software development teams across the world. It is also known as an agile management tool which supports issue tracking, sprint planning, backlog management, customizable workflows, detailed reporting, user role administration and integration with several development and DevOps tools.

The primary goal of this project is to validate Jira's core functionality to make sure it functions as intended, satisfies the functional requirements and offers smooth and effective user experience for a range of roles, including developers, project managers and administrators.

Our testing scope is limited to 10 core features to verify their efficiency towards day-to-day project and issue management. The testing will include verifying the user login and authentication process, issue creation, and management; the updating of issue details, commenting, status changes through workflows, search, linking of issues and creating customized forms.

Test execution will mainly be manual with potential automation. The focus will be identifying any defects at the earliest time possible, verifying functional correctness, usability and asserting Jira's reliability under real-world usage scenarios.

Test reports will give clarity to the stakeholders in their decision-making process on whether Jira is ready to be released and endorsed by users, thus upholding the existing quality standards and quality enhancement of the product.

2. TESTING SCOPE AND OBJECTIVES

2.1 Scope:

The focus is on 10 core features of daily project and issue management:

1. User Login
2. Creating new issues
3. Assigning issues
4. Updating issue fields

5. Adding comments to issues
6. Changing issue status through workflow transitions
7. Searching for issues
8. Viewing issue details
9. Linking related issues
10. Create Form to gather information

2.2 Objectives:

- To verify that users can successfully log in and authenticate with valid credentials and are appropriately handled when credentials are invalid.
- To ensure that users can create, assign, update, and manage issues with accurate field validations.
- To validate that comments and issue statuses update correctly and reflect immediately.
- To confirm that search functionality returns relevant results efficiently.
- To check that issue linking works properly and related information displays as expected.
- To verify that the forms collect information correctly
- To assess the usability of these features ensuring users can perform tasks intuitively.
- To detect and report any functional defects or usability issues that might impact the user experience.

3. ASSUMPTIONS & RISKS

3.1 Assumptions:

- ✓ Jira Cloud is already developed and deployed and accessible for testing.
- ✓ All API connections and integrations needed for testing work properly.
- ✓ The features and requirements will not change during the testing period.
- ✓ Test user accounts with correct permissions are available.
- ✓ Test data like projects and issues will be ready or can be created.

- ✓ The testing tools, browsers and internet connection will work without problems.

3.2 Risk:

- Jira Cloud might have updates or maintenance that could stop testing.
- External integrations may cause failures outside tester control.
- Changes in features or unclear specs during testing period might cause incorrect tests.
- Limited permissions may restrict full testing coverage.
- Insufficient test data could result in testing incompleteness.
- Network issues could cause delays or failures during testing.

4. TYPES OF TESTING

- **Functional Testing:**

To verify that all core Jira features work as expected from an end-user perspective.

- **Usability Testing (Non-functional):**

To ensure the Jira user interface is intuitive, easy to navigate, and accessible for users with different roles.

- **Performance Testing (Non-functional):**

To evaluate Jira's responsiveness and stability during normal user load and typical usage conditions.

- **Security Testing (Non-functional):**

To validate that user roles, permissions, and access controls are properly enforced to protect sensitive information.

- **API Testing**

To verify that Jira's API endpoints related to issue and user management respond correctly and handle requests reliably.

5. TEST APPROACH & METHODOLOGY

- Testing will be primarily manual, with exploration based on real user scenarios.
- Test cases will be designed from the perspective of different user roles (Admin, Developer, Viewer).
- Automation will be implemented for one critical feature (Issue Tracking) using Selenium WebDriver with JavaScript.
- Defects will be logged and tracked using Jira itself for traceability.
- Frequent communication with development for clarification and issue resolution.

6. TEST ENVIRONMENT

Application Under Test	Jira Cloud (web-based)
Browsers	Google Chrome
	Opera
	Safari
	Mozilla Firefox
	Microsoft Edge
Operating System	Windows 11
	macOS Catalina or later
	Ubuntu 22.04 or later
Testing Tools	Manual testing using web browsers
	Selenium WebDriver with JavaScript (Node.js) for automation
	LoadRunner for Performance testing
	Postman for API testing
Network	Stable internet connection with adequate bandwidth
Test Data	Test user accounts with necessary permissions, pre-created projects and issues or ability to create them during testing

7. TEST ENTRY/EXIT CRITERIA

7.1 Entry Criteria:

- ✓ The Jira Cloud environment is accessible and stable.
- ✓ Test user accounts with required permissions are created and ready.
- ✓ Test data such as projects, issues, and workflows are set up or can be created during testing.
- ✓ Test cases for all 10 features are designed, reviewed, and approved.
- ✓ Testing tools (browsers, LoadRunner, Postman and Selenium setup) are installed and configured.
- ✓ No critical blocking issues are present in the system that would prevent testing.
- ✓ Requirements and feature specifications are finalized and will not change during the test execution.

7.2 Exit Criteria:

- ✓ All planned test cases have been executed.
- ✓ All critical and high priority defects identified during testing have been reported and documented in the bug tracking system.
- ✓ No critical or major defects remain open without a valid reason.
- ✓ The test results are documented and reviewed by stakeholders.
- ✓ Automation scripts (for the selected feature) have been completed and validated.
- ✓ Test summary report is finalized and signed off.

8. ROLES AND RESPONSIBILITIES

Responsibilities Roles	QA Engineer	Project Manager	Dev Team	Client/ Stakeholder
Requirement Analysis	✓			
Test Planning	✓			
Test Case Design	✓			

Manual Test Execution	✓			
Defect Reporting	✓			
Automation Scripting	✓			
Test Progress Reporting	✓			
Coordinating with the Development Team for clarifications	✓		✓	
Review and Approval of Test Deliverables		✓		✓
Final Sign-off				✓

9. TOOLS TO BE USED

Manual Testing Tools:

- Web browsers Google Chrome, Mozilla Firefox, Opera and Microsoft Edge for manual testing of Jira Cloud features.

Automation Tools:

- Selenium WebDriver with JavaScript (Node.js) for automating Jira features.
- VS Code for writing and maintaining automation scripts.
- Postman for automated API testing of Jira endpoints.

Defect Tracking Tools:

- Jira/Bugzilla bug tracking system to log and manage bugs.

10. TEST SCHEDULE

The schedule assumes an 8-hour workday

