



GOLDFINGER
H O L D I N G S

Ten Commandments for Choosing a Test Management System for Jira



When it comes to issue tracking, it is a popular notion that there should be a formally recorded correlation between requirements and defects. This correlation provides full traceability of the recorded defects and requirements and, that in turn, helps you in defect validation. Most issue tracking systems, including Jira, enable you to link defects and requirements. However, in addition to this, you should also be able to link the test cases with the defects and requirements. This is called the traceability triangle.

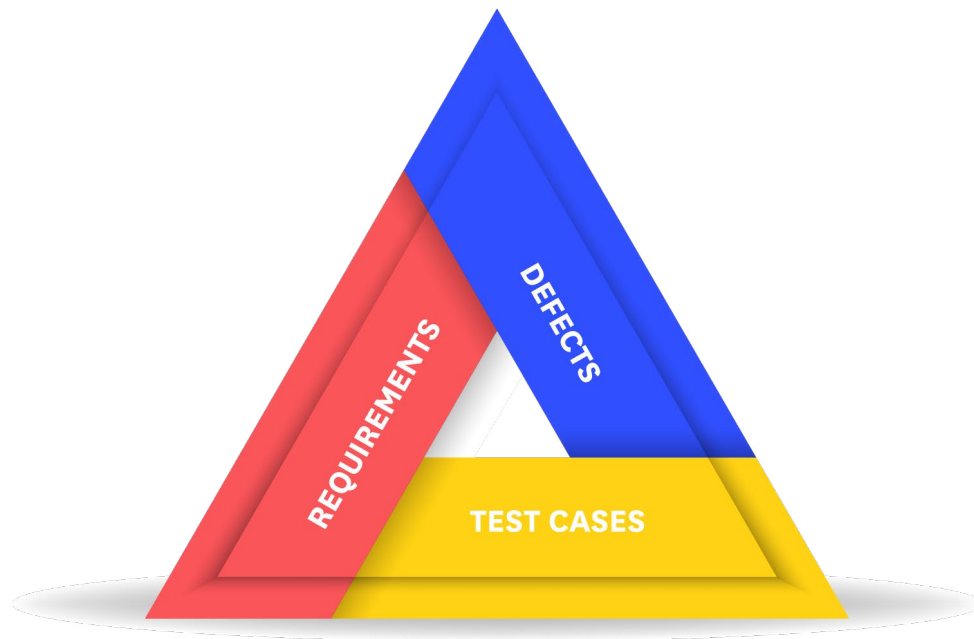


Figure 1: Traceability triangle

Why is it essential to trace test cases? Now, assume that the test cases are not linked to the defects and

requirements. All you will submit to your customers are the defects they reported, along with the status of the test results. This will add no meaning to them. Test cases give you a complete test coverage report of what was tested, how it was tested, and what the result was. This serves as a reference for your team and an evidence for the customers. Jira allows you to link defects and requirements out-of-the-box. However, there's no option to link the test cases.

This is why you need a test case management system. There are a plethora of test case management solutions for Jira. Nevertheless, before making a choice, you need to factor in a host of things. Here we discuss the ten important considerations to take into account when choosing a test management system for Jira:

I. The test management tool should be INTEGRATED with Jira

There are quite a lot of test management tools that are connected to Jira. But, there are certain downsides of choosing a tool that is just connected to Jira, but not integrated with it:

- If you have two different applications, you need to switch back-and-forth between Jira and the test management tool

- Since the test management tool is not integrated with Jira, every time you make some changes, you need to ensure it gets reflected in Jira
- There's a learning curve associated with getting familiar with the interface of a new test management tool that does not follow Jira design guidelines

Choosing a test management tool that is integrated with Jira helps you overcome these challenges, since all the elements are available as an integral part of Jira, unlike a tool that is merely connected to Jira.

2. Test cases should be a part of the agile board

Most engineering teams have taken an agile approach to software development. Typically, during the sprint planning, the development team takes into account the time taken to develop a feature, while the time taken by the test engineers to test the output is not considered. This is why it is important to choose a test management system that includes the test cases in the agile board so developers can get a clear picture of the complexity of the test cases and take that into consideration while planning a sprint. And, this gives the test engineers a sense of satisfaction, while it also helps the development team plan the sprint more accurately.

3. The tool should allow you to group test cases hierarchically based on functional decomposition model

The test management tool you choose should allow you to define the functionality of your application in a hierarchical manner and attach test cases for each functional area. The grouping of test cases based on functional areas is called a test suite. Test suites allow you to create test plans based on functional areas and also help you identify the functionalities that fell through the cracks while devising your test plan. For example, if you want to do a login smoke test, you should be able to subsume the test plan under that particular test suite. Every critically viable test case management system must support test suites.

4. The tool should allow you to view test cases from a requirement point of view

Let's say there's a user story and you want to verify if the test cases are congruous with the story. In this case, it's essential to be able to view your test cases from within the user story. Therefore, make sure that the test management tool lets you associate the test cases with the user story and vice versa so you can easily validate the test cases from a user story perspective. This helps you understand if the test cases are aligned with the requirements.

Experimental Project For QA Only / RP-151
A story for Atlassian Summit 2016

Edit Comment Assign More Start Progress Admin

Type: Story Status: OPEN (View Workflow)
Priority: High Resolution: Unresolved
Component/s: None Fix Version/s: None
Labels: None

Description
This is my exciting presentation.

Requirements

Create Parent Create Child Link Parent Link Child Requirement Tree

- RP-157 This epic is a parent for the story OPEN
- RP-151 A story for Atlassian Summit 2016

Test Cases

View Bugs Create Test Case Link Test Case

- RP-152 A test for Atlassian Summit 2016 OPEN
- RP-156 Another test case for Atlassian Summit 2016 OPEN

Figure 2: Test coverage for a story

Image source: softwaretestingmagazine.com

5. The tool should enable you to reuse the test plans in different environments

For instance, let's consider you want to execute a test case in different environments like Chrome, Safari, and Firefox. You may first execute it in Chrome and update the result. What if you execute it subsequently in Safari and update the result once again? It will override the existing results and hence, you will lose the previous result. This is why you need a test management tool that allows you to reuse the test plans. An ideal solution is one that lets you create instances of test plans called test cycles. These test cycles are independent of test plans and can be reused for different purposes, while the test plan will act as a blueprint. You can view the results of the test cycles under the respective test plan.

6. The tool should provide an aggregate view of test plan results

Individual reports on test cases, test plans, and test cycles can give you granular details of the tests. However, unless you get an aggregate view, you'll not be able to see the bigger picture of what's happening. It is, therefore, important for the test management tool to give you a consolidated view of the test cases, test cycles, and requirements. This will give your manager or test lead a comprehensive view of the execution of test plans.

7. The tool should provide a traceability matrix

The traceability matrix is another essential feature that you should look for in a test management tool. This matrix can be represented in the form of a table, giving you a comparison of the test results from sprint to sprint. It, therefore, helps the manager assess if the team maintains consistency in the quality of the output.

8. The tool should provide configurable gadgets on the user's dashboard

This is an extremely useful feature for high-level assessment of product quality. If the tool offers you the flexibility to customize your dashboard with the help of configurable gadgets, you can track the progress of your project in a granular manner.

synapseRT Test Plan

Project: ProtectV Release: 1.2

Test Plan	In Progress	Passed	Failed	Can't Test
Linux Kernel Upgrade Test plan for vSphere Cloud	0	6	1	2
Linux Kernel Upgrade Test plan for AWS Cloud	0	7	0	0
Test Plan - Test Plan for Windows Upgrade Support...	0	13	0	0
ProtectV-1.2_Regression test	0	194	4	1
ProtectV-1.2_Release test	0	30	0	0
Test Plan:ProtectV CentOS 5.7 (64bit) VMs, Cloud=...	0	41	0	1
Test Plan for Linux Upgrade Support for 1.2.0	0	5	0	0
Test Plan:ProtectV CentOS 5.7 (64bit) VMs, Cloud ...	0	47	2	0
ProtectV Client Uninstallation	7	0	0	0
Release 1.2 ECO test plan	0	35	0	0
CLONE - ProtectV Client Uninstallation - TEST	0	6	1	0
Total	7	384	8	4

Figure 3: An example of a configurable gadget

Image source: softwaretestingmagazine.com

9. The tool should support test automation

Gone are those days when testing was synonymous with manual testing. In the past few years, automated testing has picked up steam. Hence, it is important for your test management tool to support automation capabilities. Specifically, the tool you use must have a RESTful API that lets you update the test case results in Jira programmatically

10. The tool should enable you to run automated tests and view the results from within Jira

Test plans consist of both automated and manual tests. The automated tests are typically executed via a CI server like Jenkins. If your test management solution offers you the flexibility to trigger execution of automated tests on the CI server (such as Jenkins) directly from Jira, it will help people without coding knowledge run tests automatically with the click of a button. The job on Jenkins will be started, executed, and the results updated in Jira.

If you're planning to enhance the capabilities of Jira with a test management system, make sure that the solution you choose meets all the requirements discussed above. However, it is hard to find a third-party testing tool that supports all these capabilities. Hence, it is important to look for an application that is built exclusively for Jira.

synapseRT by **Goldfinger** is a holistic test management solution that is built specifically for and tightly integrated with Jira. The tool is designed to extend Jira's use for end-to-end test management so you can plan and manage all aspects of quality testing right inside Jira – without having to choose an additional testing tool. This results in tighter collaboration between your product management, development, and testing teams within the same environment. As a result, your teams can move significantly faster, identify problems earlier and with greater accuracy, and release higher quality software.

Do you have more questions about synapseRT? [Contact us](#) to schedule a demo.





GOLDFINGER
H O L D I N G S



Goldfinger Holdings is an industry-leading technology company dedicated to helping organizations extend the capabilities of their mission-critical applications to power innovation, deliver value, and drive growth. We offer a comprehensive suite of solutions – ranging from value stream integration, to agile test management, to federal identity and access management – to introduce enhanced levels of visibility, quality, and security into the software development lifecycle.

Goldfinger’s synapseRT for Jira is a test management solution that streamlines software testing in Agile and DevOps environments so your teams can test faster, improve software quality, and deploy more rapidly and efficiently. For more information, visit www.goldfingerholdings.com or follow us on [LinkedIn](#), [Twitter](#), and [Facebook](#).