

# Xporter Template: Test Report

- [Purpose](#)
- [Output Example\(s\)](#)
- [How to use](#)
  - [Source data](#)
  - [Output format](#)
  - [Report assumptions](#)
  - [Usage examples](#)
    - [Export the test specification of all Tests having the "Login" component, or by other criteria based on the Test issue fields](#)
    - [Export the test specification for all Tests contained in a Test Repository folder](#)
    - [Export the test specification for all Tests contained in a Test Plan or in one of its folders \(including sub-folders\)](#)
- [Understanding the report](#)
  - [Layout](#)
    - ["Introduction" section](#)
      - [Document Overview](#)
      - [Total tests](#)
      - [Requirements covered by these Tests](#)
    - [Tests](#)
      - [Test Summary](#)
      - [Test Description](#)
      - [Test Issue Attachments](#)
      - [Status of the last execution](#)
      - [Requirements](#)
      - [Preconditions](#)
      - [Test Details](#)
      - [Test Sets](#)
      - [Test Plans](#)
    - [Appendix A: Approval](#)
  - [Customizing the report](#)
    - [Adding or removing information to/from the report](#)
      - [Exercise: add a field from the related "requirement" issue](#)
- [Performance](#)
- [Known limitations](#)

## Purpose

This report enables you to extract the detailed specification of one or more Tests, so that you can generate a document that contains a snapshot of the latest specification. This can be used for revision/approval, or even to share it with someone else that hasn't access to Jira and wishes to pursue testing.

Possible usage scenarios:

- see the detailed specification of the Test(s), including the test steps of manual scripted test cases and the Gherkin of BDD test scenarios
- see all Preconditions required by the Test(s)
- see all the requirements covered by the given Test(s)
- see all Test Sets where the Test(s) are included in
- see all Test Plans where the Test(s) are included in

## Output Example(s)

The following screenshots shows an example of the sections you should expect in this report.



# TEST REPORT

Project Name: Essentials Workshop BCK

Prepared by: |

CONFIDENTIAL

Document Date: 12/12/2022



## 2. Tests

### 2.1. EWB-271: Validate country and locale selectors.


#### 2.1.1. Test Summary

Test Type	Label(s)	Component(s)	Priority
Cucumber	Cucumber, Testing	Login, Modules	Medium

#### 2.1.2. Test Description

Validate that the different options available for country and locale are working as defined in the requirement.

#### 2.1.3. Test Issue Attachments

File Name	Author	File Size (bytes)
<a href="#">UCI ARRABIDA 20.pdf</a>	Sérgio Freire	220301
<a href="#">locale_selection.png</a> 	Sérgio Freire	62051

#### 2.1.4. Status of the last execution

FAILED


#### 2.1.5. Requirements

Key	Summary	Version	Workflow Status
<a href="#">EWB-270</a>	Build login page		To Do

TEST REPORT – AUTOMATICALLY GENERATED	5
---------------------------------------	---

## 2.1.23. Test Details

This Manual test has 6 steps.

Step	Action	Data	Expected Result	Attachments
1	Access endpoint of the application		Main page is loaded	<a href="#">Screenshot 2022-11-28 at 15.05.44.png</a>  <p>Project Name: Bookstore Item: BOOK-31 Prepared By: Agnès IT Admin Document Date: 28/11/2022 CONFIDENTIAL</p>
2	Click on the List owner menu entry		<u>Owners</u> page is loaded	
3	Choose on		Owner details page is	



### 3. Appendix A: Test Report Approval

The undersigned acknowledge they have reviewed the **Test Report** and agree with the approach it presents. Changes to this **Test Report** will be coordinated with and approved by the undersigned or their designated representatives.

Signature:	Date:
_____	_____
Print Name:	_____
Title:	_____
Role:	_____

## How to use

This report can be generated from the Issue details screen, or from the Issue Search/navigator page.



### Learn more

General information about all the existing places available to export from, and how to perform it, is available in the [Exporting](#) page.

## Source data

This report is applicable to:

- 1 or more Test issues

## Output format

The standard output format is .DOCX, so you can open it in Microsoft Word, Google Docs, and other tools compatible with this format. You can also export it to .PDF.

## Report assumptions

The template has a set of assumptions that you make sure that your Jira/Xray environment complies with:

1. Issue types having these names
  - a. "Test", "Test Set", "Test Plan"
2. Tests use the "tests" issue link type to cover issues (e.g., Story)

If any of these assumptions is not met, you need to update the template accordingly.

## Usage examples

### Export the test specification of all Tests having the "Login" component, or by other criteria based on the Test issue fields

1. open the Issue Search/navigator (go to **Filters > Advanced issue search**)
2. search by component "Account", or by other criteria based on the Test issue's fields

The screenshot shows the Jira Software Search page. The search filters are set to: Project: All, Test, Status: All, Assignee: All, Contains text, and Component: Account. The search results table lists several test issues, including BOOK-517 (Manual Login), BOOK-498 (Manual Test), BOOK-309 (CanSubtract), BOOK-37 (Manual test of password reset procedure), BOOK-36 (Manual test of strong password validation), BOOK-18 (Test a visitor can change his locale), BOOK-17 (Test a logged in visitor can edit the default address), BOOK-16 (Test a logged in visitor can edit the account details), BOOK-15 (Test logged in visitors can logout from their account), and BOOK-14 (Test visitors can login to Calculator Website).

- a. Select the "Xporter" action and export it using this template

The screenshot shows the Jira Software Search page with the 'Export' dropdown menu open. The 'Xporter' option is highlighted in red. The search filters and results table are the same as in the previous screenshot.

### Export the test specification for all Tests contained in a Test Repository folder

1. select a folder in the Test Repository
2. Select the "View Issues" action; you'll be redirected to the Issue Search/Navigator

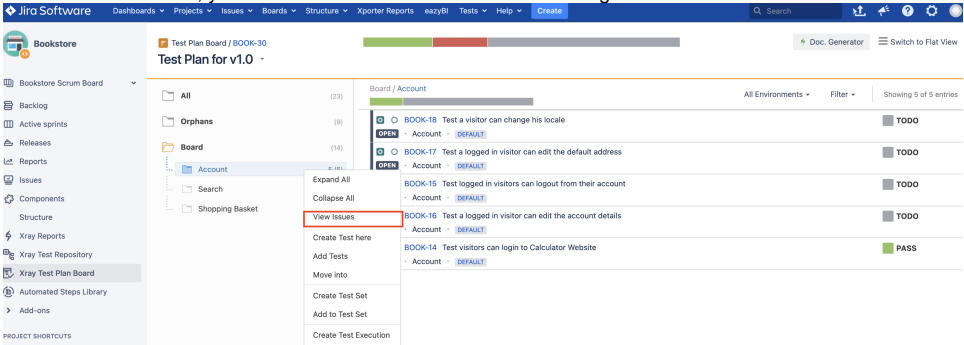
The screenshot shows the Jira Software Test Repository for project Bookstore. The 'Test Repository' folder is selected, and the 'View Issues' action is highlighted in red. The search filters and results table are the same as in the previous screenshot.

- a. The generated query will include all Tests in all sub-folders; to just include the Tests on the selected folder, change the 'true' to 'false' on the last argument of the TestRepositoryFolder() JQL function
4. On the Issue Search/Navigator, select the "Xporter" action and export it using this template

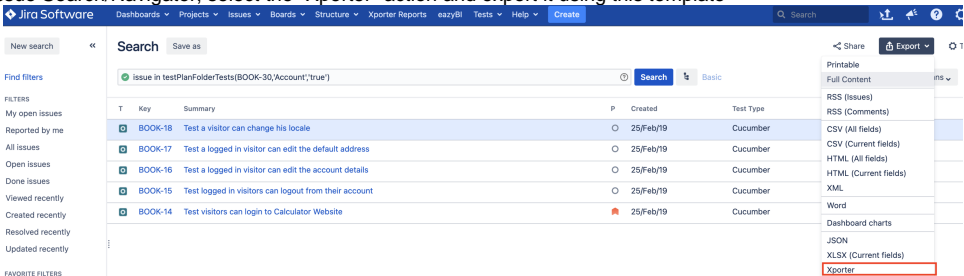
The screenshot shows the Jira Software Search page with the 'Export' dropdown menu open. The 'Xporter' option is highlighted in red. The search filters and results table are the same as in the previous screenshot.

Export the test specification for all Tests contained in a Test Plan or in one of its folders (including sub-folders)

- 1. select a folder in the Test Plan
- 2. Select the "View Issues" action; you'll be redirected to the Issue Search/Navigator



- a.
- 3. The generated query will include all Tests in all sub-folders; to just include the Tests on the selected folder, change the 'true' to 'false' on the last argument of the testPlanFolderTests() JQL function
- 4. On the Issue Search/Navigator, select the "Xporter" action and export it using this template



Understanding the report

The report shows detailed information about the Tests, focused on their specification (e.g., manual steps, Gherkin, test charter).

Layout

The report is composed by several sections. Two major sections are available: Introduction and Test Executions details.

By default, and to avoid overload/redundancy of information, only the "Introduction" section will be rendered; you can change this behavior on the template (more info ahead).

"Introduction" section

This section is divided into 3 sub-sections:

- 1. Document Overview
- 2. Total tests
- 3. Requirements covered by these Tests

Each of these sections is explained below.

Document Overview

Brief description of what you will find in this report and how it was generated.

Total tests

In this section, the total tests are grouped by their test type and a count is shown.

Test Type	Number of Tests
Manual	Number of tests that are step/script based; usually these are known by "manual" (even though "manual" is a rough simplification)

Gherkin	Number of Gherkin based tests; usually these correspond to Cucumber test scenarios, with Test Type "Cucumber", but may also refer to other Gherkin based frameworks
Other	Number of tests that are neither scripted nor Gherkin based; usually these correspond to unstructured tests; this includes the tests having Test Type "Generic". Generic automated test scripts are usually included here; exploratory tests also.



#### Please note

If you wish to count Tests by custom Test Types, you can do so. See more information ahead on how to customize this report.

The output will have the following information.

## 1.2. Total tests

This is the current distribution of tests by their type.

Test Type	Number of Tests
Manual	1
Gherkin	1
Other	0

## Requirements covered by these Tests

In this section we have an overview of all the requirements that are covered by the Tests in this report.

Field	Description	Code
Key	Key of the requirement (in this case we are adding it as link)	@{title=\${Links[j].Key}} href=\${BaseURL}/browse/\${Links[j].Key}}
Summary	Summary of the requirement	\${Links[j].Summary}
Version	Project version assigned to the requirement	\${Links[j].FixVersions}
Workflow Status	Workflow Status of the requirement	\${Links[j].Status}

The requirements are listed in a table as follows.

## 1.3. Requirements covered by these Tests

Key	Summary	Version	Workflow Status
<a href="#">EWB-270</a>	Build login page		To Do

## Tests

This section will gather all the information related to each Test, focused on its specification.

It is composed with several sub-sections that will be filled with information if it is available or be filled with a message showing that no information is available.

### Test Summary

This section have a table with information regarding each Test in this report (and will repeat these sections for each Test). The information is presented as a table with the following fields:

Field	Description	Code
-------	-------------	------

<i>Test Type</i>	Test Type	<i>\${Test Type}</i>
<i>Test Repository Folder</i>	Folder where Test is at in the Test Repository	<i>\${TestRepositoryPath}</i>
<i>Label(s)</i>	Labels assigned to the Test issue	<i>\${Labels}</i>
<i>Component(s)</i>	Components assigned to the Test issue	<i>\${Components}</i>
<i>Priority</i>	Priority assigned to the Test issue	<i>\${Priority}</i>

The resulting table look like the one below.

## 2. Tests

### 2.1. [EWB-271](#): Validate country and locale selectors.

#### 2.1.1. Test Summary

Test Type	Label(s)	Component(s)	Priority
Cucumber	Cucumber, Testing	Login, Modules	Medium

#### Test Description

The description of the Test (*\${wiki:Description}*).

#### 2.1.2. Test Description

Validate that the different options available for country and locale are working as defined in the requirement.


#### Test Issue Attachments

This section only appears if you have any attachments associated to the Test.

Key	Description
File Name	File Name of the attachment
Author	The Author of the attachment
File Size	File Size of the attachments in bytes

This appears in the document in a table form. A thumbnail is shown for images.

2.1.3. Test Issue Attachments

File Name	Author	File Size (bytes)
<a href="#">UCI ARRABIDA 20.pdf</a>	Sergio Freire	220301
<a href="#">locale_selection.png</a> <div><div>EN</div></div>	Sergio Freire	52051

Status of the last execution

This section shows the last execution status for the Test.

2.1.2. Status of the last execution

FAILED

Requirements

In this section we have an overview of all the requirements that are covered by the Test.

Field	Description	Code
Key	Key of the requirement (in this case we are adding it as link)	@{title=\${Links[j].Key} href=\${BaseURL}/browse/\${Links[j].Key}}
Summary	Summary of the requirement	\${Links[j].Summary}
Version	Project version assigned to the requirement	\${Links[j].FixVersions}
Workflow Status	Workflow Status of the requirement	\${Links[j].Status}

The requirements are listed in a table as follows.

2.1.5. Requirements

Key	Summary	Version	Workflow Status
<a href="#">EWB-270</a>	Build login page		To Do

Preconditions

If the Test requires any Preconditions, they will be shown.

Key	Description	Code
Key	Key of the Precondition in the form of a link	@{title=\${PreConditions[n].Key} href=\${BaseURL}/browse/\${PreConditions[n].Key}}
Summary	Summary of the Precondition	\${PreConditions[n].Summary}
Condition	Condition	\${wiki:PreConditions[n].Conditions}






The Preconditions are listed in a table as follows.

2.3.6. Preconditions

Key	Summary	Condition
<a href="#">EWB-242</a>	User is authenticated	Access login page  Authenticate using a valid user

Test Details

This section shows the Test details; for that we are considering the different possible Test Types we can have in Xray. For each type we will fetch different information.

Type	Key	Description	Code	Output																																				
Unstructured  (e.g., "Generic")	Test Type	Test Type field	<code>\${TestType}</code>	<b>2.2.7. Test Details</b> <table><tr><td>Test Type</td><td>Generic</td></tr><tr><td>Specification</td><td>login.spec.ts Login validations.Login validations Login with valid credentials</td></tr></table>	Test Type	Generic	Specification	login.spec.ts Login validations.Login validations Login with valid credentials																																
	Test Type	Generic																																						
Specification	login.spec.ts Login validations.Login validations Login with valid credentials																																							
	Specification	Definition of the Generic test	<code>\${GenericTestDefinition}</code>																																					
Gherkin  (e.g., "Cucumber")	Test Type	Test Type field	<code>\${TestType}</code>	<b>2.1.7. Test Details</b> <table><tr><td>Test Type</td><td>Cucumber</td></tr><tr><td>Gherkin Specification</td><td>Given I am logged in And I navigate to the locale selector page When I select &lt;country&gt; as a country Then The &lt;locale&gt; site is shown  Examples: <div>  country   locale     USA   US     Portugal   PT     Spain   ES     Germany   DE  </div></td></tr></table>	Test Type	Cucumber	Gherkin Specification	Given I am logged in And I navigate to the locale selector page When I select <country> as a country Then The <locale> site is shown  Examples: <div>  country   locale     USA   US     Portugal   PT     Spain   ES     Germany   DE  </div>																																
	Test Type	Cucumber																																						
Gherkin Specification	Given I am logged in And I navigate to the locale selector page When I select <country> as a country Then The <locale> site is shown  Examples: <div>  country   locale     USA   US     Portugal   PT     Spain   ES     Germany   DE  </div>																																							
	Gherkin Specification	Gherkin specification of the Test	<code>\${CucumberScenario}</code>																																					
Structured  (e.g., "Manual")	(Step)	Step Number	<code>\${TestSteps[m].StepNumber}</code>	<b>2.1.7. Test Details</b> <p>This Manual test has 4 steps.</p> <table><tr><th>Step</th><th>Action</th><th>Step Custom Fields</th><th>Data</th><th>Expected Result</th><th>Attachments</th></tr><tr><td>1</td><td>Access the Calculator application</td><td></td><td></td><td>The login page appears</td><td><a href="#">login.png</a> </td></tr><tr><td>2</td><td>Insert <code>\${UserName}</code> in the <code>User Name</code> field</td><td></td><td></td><td>The <code>\${UserName}</code> appear in the <code>User name</code> form field</td><td></td></tr><tr><td></td><td>Action</td><td>Action of the Test Step</td><td><code>\${TestSteps[m].Action}</code></td><td></td><td></td></tr><tr><td></td><td>Step Custom Fields</td><td>Step custom field(s)</td><td><code>\${TestSteps[m].&lt;step_custom_field&gt;}</code></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td><i>Note: Test Step Custom Fields are not rendered by default; they need to be specified explicitly</i></td><td></td><td></td></tr></table>	Step	Action	Step Custom Fields	Data	Expected Result	Attachments	1	Access the Calculator application			The login page appears	<a href="#">login.png</a> 	2	Insert <code>\${UserName}</code> in the <code>User Name</code> field			The <code>\${UserName}</code> appear in the <code>User name</code> form field			Action	Action of the Test Step	<code>\${TestSteps[m].Action}</code>				Step Custom Fields	Step custom field(s)	<code>\${TestSteps[m].&lt;step_custom_field&gt;}</code>						<i>Note: Test Step Custom Fields are not rendered by default; they need to be specified explicitly</i>		
	Step	Action	Step Custom Fields		Data	Expected Result	Attachments																																	
	1	Access the Calculator application				The login page appears	<a href="#">login.png</a> 																																	
2	Insert <code>\${UserName}</code> in the <code>User Name</code> field			The <code>\${UserName}</code> appear in the <code>User name</code> form field																																				
	Action	Action of the Test Step	<code>\${TestSteps[m].Action}</code>																																					
	Step Custom Fields	Step custom field(s)	<code>\${TestSteps[m].&lt;step_custom_field&gt;}</code>																																					
			<i>Note: Test Step Custom Fields are not rendered by default; they need to be specified explicitly</i>																																					

Data	Data of the Test Step	<code>\${TestSteps[m].Data}</code>
Expected Result	Expected Result of the Test Step	<code>\${TestSteps[m].ExpectedResult}</code>
Attachments	Attachment of the Test Step; images will be rendered	<pre>@ {title=\${TestSteps[m].Attachments[s].Name}  href=\${TestSteps[m].Attachments[s].FileURL} }} ! {\${TestSteps[m].Attachments[s].FileURL} }  width=100}</pre>

## Test Sets

If the Test is part of Test Sets, they will be shown.

Key	Description	Code
Key	Key of the Test Set in the form of a link	<code>@{title=\${JQLIssues[y].Key} href=\${BaseURL}/browse/\${JQLIssues[y].Key}}</code>
Summary	Summary of the Test Set	<code>\${JQLIssues[y].Summary}</code>
Version	Project version assigned to the Test Set	<code>\${JQLIssues[y].FixVersions}</code>

The Test Sets are listed in a table as follows.

### 2.3.8. Test Sets

Key	Summary	Version
<a href="#">EWB-204</a>	My <u>TestSet</u> - Validate Order	
<a href="#">EWB-217</a>	My Execution - Order Validation Test Set order	

## Test Plans

If the Test is part of Test Plans, they will be shown.

Key	Description	Code
Key	Key of the Test Plan in the form of a link	<code>@{title=\${JQLIssues[x].Key} href=\${BaseURL}/browse/\${JQLIssues[x].Key}}</code>
Summary	Summary of the Test Plan	<code>\${JQLIssues[x].Summary}</code>
Version	Project version assigned to the Test Plan	<code>\${JQLIssues[x].FixVersions}</code>

The Test Plans are listed in a table as follows.

### 2.3.9. Test Plans

Key	Summary	Version
<a href="#">EWB-8</a>	Sprint V1.2 Tests	1.2
<a href="#">EWB-237</a>	Regression Tests (DEMO)	
<a href="#">EWB-338</a>	Validate sprint 1.2 developments	

## Appendix A: Approval

This section is added for the cases where you need to have a signature validating the document.



### 3. Appendix A: Test Report Approval

The undersigned acknowledge they have reviewed the **Test Report** and agree with the approach it presents. Changes to this **Test Report** will be coordinated with and approved by the undersigned or their designated representatives.

Signature: \_\_\_\_\_ Date: \_\_\_\_\_  
Print Name: \_\_\_\_\_  
Title: \_\_\_\_\_  
Role: \_\_\_\_\_

## Customizing the report

### Adding or removing information to/from the report

To add additional information, usually we're thinking of adding:

- fields of the Preconditions, "requirements", Test Sets, or of the Test Plans associated with the Test

### Exercise: add a field from the related "requirement" issue

Let's say we have a "Severity" field on the "requirement" (i.e., covered issue) that is linked to the Test, and that we want to show it on the report.

We can copy the column "Summary" from the "Requirements" section and adapt it.

1. on the "Requirements" section,
  - a. copy "Summary" (i.e., insert a column next to it and copy the values from the existing "Summary" column)
  - b. change the contents to
    - i. `${JQLIssues[z].Summary}`

# Performance

Performance can be impacted by the information that is rendered and by how that information is collected/processed.



## Some tips

- Remove rendering of images for Test Issue Attachments and for Test Step's attachments
- limit the number of input issues

# Known limitations

- modular and data-driven tests are not supported