

Roadmap



Xray's roadmap is constantly reviewed and redefined. We update it often depending on the feedback we receive from our clients and internal stakeholders.

Our release plan is available [in our Jira issue tracker](#), although account registration is required. Feel free to vote on the issues that you would like to see implemented.

Here you can find a list of features that define our main goals for future releases. This doesn't mean that other, potentially smaller features, won't be implemented as well.

Shipped

V4.0

The Test Steps component was the target of a major facelift with the goal of providing a wider, more usable UI that will have the ability to edit all step-related fields at once, in a grid or column-based layout.

Editing, reading, and navigating through Test steps are now much easier with this new UI.

Test Steps dialog

+ Add Step

Action*	Data	Expected Result
1 Action* <div style="margin-top: 5px;"> Style ▾ B I U A ▾ A ▾ 🔍 ▾ 📏 ▾ 🗑️ ▾ 🔄 ▾ ⌂ ▾ + ▾ </div> <div style="margin-top: 5px;"> Open the Change Password screen by selecting the option "My Profile" </div> <div style="margin-top: 10px; background-color: #fff; border: 1px solid #ccc; padding: 5px;"> User mention Table Code Preformatted Panel Horizontal ruler Speech recognition Text help Visual help </div> <div style="margin-top: 5px;"> Visual Text </div> <div style="margin-top: 5px;"> Attachments (0) </div>	Data Enter value...	Expected Result Enter value...
2 Action Fill the password fields with data	Data Current Password: p4ssw0rd New Password: p4ssw0rd Confirm New Password: p4ssw0rd	Expected Result Error: "Current password is incorrect"
3 Action Close error message and fill again the password fields with data	Data Current Password: P4ssw0rd New Password: password Confirm New Password: password	Expected Result Error: "New password is too simple"
4 Action Close error message and fill again the password fields with data	Data Current Password: P4ssw0rd New Password: P4ssw0rd Confirm New Password: P4ssw0rd	Expected Result Information message: "Password successfully changed"

Along with the Test steps UI revamp, Xray also provides the ability to configure and specify new custom step fields for manual test cases that can complement the standard ones (Action/Step, Data, Expected Result). The standard custom fields can also be hidden if desired. All of this can be configured in the project settings.

Test Steps dialog

Grid

+

Add Step

↓

Attachments (0)

↑

9

Execution Domain

BACKOFFICE

Action

Click on the **Execute** order button.

Data

None

Expected Result

The order status must now be **Processing**.

↓

Attachments (0)

↑

10

Execution Domain

RDBMS

Action

Execute the following database script.

Data

SELECT * FROM ORDERS WHERE ID = <ORDER_ID>

Expected Result

This must return the database order row with the following information:

- Status: "PROCESSING"
- Source: "Backoffice"

↓

Attachments (0)

↑

11

Execution Domain

POSTMAN

Action

Execute the following request.

Data

https://staging-server.local:8662/store/order/<ORDER_ID>

Expected Result

The following response must be returned:

```
{  "orderId": "<ORDER_ID>",  "status": "PROCESSING",  "source": "Backoffice",  "items": [    ...  ]}
```

↓

Attachments (0)

↑

12

Execution Domain

Backoffice

Action

Click on **Finish** to close the order.

Data

Enter value...

Expected Result

The order status will now be **Resolved**.
The client receives an email with the package information from the carrier.

Attachments (0)

Done Cancel

Starting from v4.0, It is possible to define additional Test Run custom fields. These fields can be useful to add extra information to Test Runs, usually only available during or after executing Tests.

Test Run custom fields can be configured by project and by Test Type. Therefore, these settings will not affect other projects within your Jira instance. For example, it is possible to have custom fields just for Manual Tests within a project.

Reporting

The Test Runs List report provides an additional column that displays the Test Run Custom Field values for each Test Run.

It is also possible to include these fields on the Tests data-table within the Test Execution issue.

Learn more about this feature [here](#).

Book Store / Test Execution: STORE-44 / Test: STORE-43

Load testing checkouts

Export Test as TextReturn to Test ExecutionNext

Execution Status **FAIL**

Assignee: Admin

Version: v1.0

Started On: 01/Jun/20 2:30 PM

Executed By: Admin

Revision: 94875gbf8qun23f

Finished On: 01/Jun/20 2:30 PM

Tests -

environments:

Comment

Preview Comment

Click to add comment

Execution Defects (0)

Create DefectCreate Sub-TaskAdd Defects

No defects yet...

Execution Evidence (1)

Add Evidence

screenshot-1.png66 kBJust now

Execution Details

Test Description

None

Test Issue Links (1)

tests

STORE-31As a visitor, I can Checkout Items in my basket

TO DO

Custom Fields

Avg CPU usage*Requests per secondError rate*

Major 60% - 80%164.3

Test Details

Test Type: Generic

Definition: jmeter.bookstore.checkout

Activity

In the works

The [Xray Connector app for Bamboo](#) will be updated in order to support new functionality and APIs already available in Xray. The Xray Connector will also support Xray cloud APIs and connectivity.

Ability to call an existing Test from the step of another Test and thus use tests as reusable building blocks for the composition of more complex testing scenarios.

The called test can be executed independently or be used as a step of a broader test (e.g. "login as the admin user").

Ability to define variables in the Test specification (e.g. the steps) that can be replaced by values inherited from different sources (a calling test for example). In that scenario, the called test is used as a parameterized template (e.g. "login as the specified user").

Ability to perform data-driven testing (i.e. execute the same test against multiple parameters, for different combinations of data).

Data-driven testing will be implemented using the foundations of test parameterization.

Data may be defined at multiple levels including:

- the test level (default values)
- the caller test
- the Test Plan
- the Test Execution

Data might also be imported from external sources (like CSV files).

Future Versions

Jira Datacenter supports archiving of issues and projects since v8.1. By archiving issues, Jira will remove them from the indexes. The archived issues will still be accessible in read-only mode, however, they will not appear in Jira searches. This enhances Jira performance because Jira stores less data.

The goal of this feature is to support issue archiving so that Xray issues can be archived without some of the undesirable side effects ([described here](#)). A good use case for archiving Xray issues is to archive old Test Executions, especially if continuous integration processes are used. These can quickly create many Test Execution issues while importing new execution results into Xray on a daily basis. These issues are good candidates to be archived.

Currently, the Test Plan is composed of a static list of Test cases. This means you must explicitly add the Tests to the Test Plan. If the Tests are all known and well defined when you start your Test Plan, this is ok. However, if you are working in an agile context where a Test Plan is created for a specific sprint, Tests will only be specified during the sprint and later added to the Test Plan. This process is not ideal because users might forget to add the Tests to the Test Plan.

Dynamic Test Plans can be defined with a JQL query which will be the source for Test cases. Considering the use case described above, we can define a JQL query that will get all Tests covering the requirements of a specific sprint.

This feature will cover a common scenario where manual Test cases evolve to automated Tests. In this case, the user can either change/select the Test type to see different definitions (refer to the configuration option: [Delete test definition when changing Test type](#)) or create separate Test issues to cover all the natures for the Test.

With Test natures, we plan to make it more explicit that multiple Test definitions can coexist within the same Test case. Users will also be able to set the current definition so that Test Runs are always created using the "current" definition.

Ability to provide management for test types and thus provide additional, user-defined test types besides the standard ones (i.e. Manual, Cucumber, Generic).

Currently, it is possible to create new Test types using the "Test Type" custom field. However, any custom test type (except for some pre-determined names) will default to the Generic kind in Xray.

We plan to create a custom view where users can define new Test types and also the kind for each Test type. For instance, one can create a new Test type named "SpecFlow" and choose the "Gherkin" field. This means SpecFlow Tests would display a field with Gherkin syntax highlight.

Most of Xray's native reports are tied to the project level. We plan to add multi-project support for these reports by allowing a saved filter with issues from multiple projects to be used as the report data source.

Right now, a Test Run contains the start and finish date which are set automatically, although users can also change the start date manually. However, users can not stop and start the execution as needed and the total time spent on the Test Run (finish - start) might not reflect the time spent by the user executing the Test.

We plan to provide a feature to allow users to pause and continue with the execution directly on the Test Run screen.

Most Xray settings are now global settings. This means only Jira administrators can change these settings. Also, having only global settings means that any change will affect all projects in Jira.

By moving some of the settings to the project level, each project/team can have a different configuration of Xray.