

Learn how to use versions in Jira Software

The guide to using versions in Jira Software



BY MAX REHKOPP

BROWSE TOPICS

- Agile manifesto
- › Scrum
- › Kanban
- › Agile project management
- › Product Management
- › Agile at scale
- › Software development

[Learn kanban with Jira Software](#)

[Learn how to use Epics in Jira Software](#)

[Learn how to create an agile board in Jira Software](#)

[Learn how to use sprints in Jira Software](#)

- [Learn Versions with Jira Software](#)

[Learn Issues with Jira Software](#)

[Learn burndown charts with Jira Software](#)

[Auto-create sub-tasks and update fields in Jira](#)

[How to automatically assign issues with Jira Software Automation](#)

[How to sync epics stories with Jira Software Automation](#)

[Automatically escalate overdue issues in Jira](#)

- › About the Agile Coach

[All articles](#)

Jira Versions Tutorial

In this tutorial, we'll explain how to work with versions in Jira Software.

Time:

10 minute read. Complete over 2 weeks or more.

Audience:

- You're familiar with how Scrum and/or Kanban work in Jira Software
 - You have the Administer Projects permission for all projects on your Scrum or Kanban board.
- See [Managing project permissions](#) for more info.

Prerequisite:

- You've created a Jira Software account
- You've created a Jira Software Scrum or Kanban project
- You've filled your project with issues

[Get it free](#)

PRODUCT DISCUSSED

Jira Software

The #1 software development tool used by agile teams

[Get it free →](#)

SUBSCRIBE

Sign up for more articles

Email

[Subscribe](#)

WHAT IS A VERSION IN JIRA SOFTWARE?

In Jira Software, versions represent points-in-time for a project. They help you organize your work by giving you milestones to aim for. You can assign issues in your project to a specific version, and organize your sprints around completing work in that version.

Step 1: Create a version in Jira Software

- 1 Navigate to your project.
- 2 In the project menu, click on **Releases**.
- 3 Select the **Version name** text box, enter a name, and click **Add**.
Version names are typically numeric, for example, **1.0** or **2.1.1**. You could also consider using an internal codename.

HOW MANY VERSIONS SHOULD I CREATE?

You can create as many as you think is necessary. For example, you might create several versions, to plan ahead. Or you might just have one or two versions for now.

Once you create a version, the **Affects version** and **Fix version** fields will become available on your issues.

Step 2: Add issues to version

If your project has a backlog

- 1 Navigate to the project backlog.
- 2 Open the **Versions** panel on the left.

The screenshot shows the Jira backlog interface. On the left, there's a sidebar with a 'VERSIONS' tab that has a circular highlight around it. The main content area shows a list of issues under 'Alpha Squad Sprint 11'. There are four issues listed, each with a checkbox and some text. Below this, there's a section for 'Alpha Squad Sprint 10' which is collapsed. At the bottom, there's a message 'Empty backlog is empty'.

- 3 Drag an issue to the version you want to add it to.

If your project doesn't have a backlog

- Open the issue you want to add to a version.
- Find the **Fix version/s** field, and enter the version you want to add the issue to.

WHAT'S THE DIFFERENCE BETWEEN FIX VERSION AND AFFECTS VERSION?

Affects version is the version in which a bug or problem was found. Although it can be useful for tracking problems, it isn't used very often in Jira.

Fix version is the version where you plan on releasing a feature or bugfix to customers. This field is used for release planning, monitoring progress and velocity, and is used widely in reporting. This is most likely the field you want.

Step 3: Monitoring the progress of a version

Jira Software gives you many tools you can use to check the progress of a version. We'll discuss a few of them here.

The release hub

The release hub provides a place for you to manage all your releases. It also gives you information on the status of your releases, and a breakdown of the number of issues in each version.

To navigate to the release hub:

- 1 Go to your project.
- 2 In the project menu, select **Releases**.

The screenshot shows the Jira Release Hub page. At the top, there are three quick filters: 'Released' (blue), 'Unreleased' (grey), and 'Archived' (light grey). Below the filters is a table with columns: Version, Status, Progress, Start date, Release date, and Description. There are four rows in the table:

- Version 4.0: UNRELEASED, No issues
- Version 3.0: UNRELEASED, [progress bar]
- Version 2.0: UNRELEASED, [progress bar] (mouse cursor is over this row)
- Version 1.0: RELEASED, [progress bar]

A tooltip for Version 2.0 displays the following information:

Version: Version 2.0
9 issues in total
Done
In Progress
To Do
All issues

At the bottom of the table are input fields for 'Version name', 'Start date (optional)', 'Release date (optional)', and 'Description'.

- 1 **Quick filters:** Focus on specific versions by filtering out the ones you aren't interested in.
- 2 **Version list:** Drag and drop versions to reorder them.
- 3 **Status:** Versions can have one of three statuses: Unreleased, Released, or Archived.
- 4 **Progress:** This shows how many issues have been assigned to the version, and how many are in each status.

WANT MORE INFORMATION TO HELP YOU MONITOR YOUR RELEASES?

If you [integrate Jira Software with your Bitbucket repository](#), you'll also see development information here relating to your commits, branches, and pull requests.

Release burndown chart (Scrum only)

The release burndown chart shows how much work has been completed, and the total work remaining. Burndown charts are used to predict your team's likelihood of completing their work in the time available. They're also great for keeping the team aware of any scope creep that occurs.

Note that to use the release burndown chart, you'll need to estimate your issues. For more information, [check out our guide on burndown charts](#).

To view the release burndown chart:

- 1 Navigate to your project.
- 2 In the project menu, select Reports.
- 3 Select the Release burndown.



- 1 **Release menu:** Select which release to view data for.
- 2 **Work added:** The dark blue segment shows the amount of work added to the release in each sprint. In this example, work is measured in story points.
- 3 **Work remaining:** The light blue segment shows the amount of work remaining in the release.
- 4 **Work completed:** The green segment represents how much work is completed for the release in each sprint.
- 5 **Projected completion:** The report projects how

many sprints it will take to complete the release, based on the team's velocity.

For more in-depth information, check out our [release burndown chart documentation](#).

Step 4: Complete a version

It's time to turn your team's hard work into a software release. By this stage, you should be confident that your version is ready to be released – issues are complete, code is checked in, reviewed, and merged, builds are passing, etc.

To deploy a release, you would typically release the version in Jira Software, build the release, then deploy the release to the required environment.

To complete a version

- 1 Navigate to your project.
- 2 In the project menu, select **Releases**.
- 3 For the version you want to release, select **Actions (…)** > **Release**.

On Kanban boards, you can also release all issues in the Done column as a new version directly from the board itself.

VERSIONS AND JIRA AUTOMATION

You can use automation to supercharge how you [use versions in Jira](#). See this automation rule and 100s more in the Jira Automation Template Library. [Go to library](#)

What to learn more?

For more detailed information on working with versions in Jira Software, check out our [versions documentation](#).

Have questions? [Ask the Atlassian Community](#).

SHARE THIS ARTICLE



MAX REHKOPF

As a self-proclaimed “chaos muppet” I look to agile



practices and lean principles to bring order to my everyday. It's a joy of mine to share these lessons with others through the many articles, talks, and videos I make for Atlassian

ARTICLE

Epics, Stories, Themes, and Initiatives

These popular agile methodologies help teams strike a healthy balance between structure and flexibility

[Read this article →](#)



NEXT TUTORIAL

Learn issues with Jira Software

Discover how teams can use issues to track individual pieces of work that must be completed.

[Try this tutorial →](#)

Agile Topics

Agile project management
Scrum
Kanban
Design

Software development
Product management
Teams
Agile at scale
DevOps

Sign up for more agile articles and tutorials.

Email

[Subscribe](#)



Up Next
[Learn Issues with Jira Software →](#)