

JIRA SOFTWARE:

Jira is a software application used for issue tracking and project management. The tool, developed by the Australian software company Atlassian, has become widely used by agile development teams to track bugs, stories, epics, and other tasks.

Why jira is used in testing?

Jira is one of the best open-source tools for planning and tracking in Agile methodology. Development teams use Jira for tracking bugs and projects, managing Scrums, and visualizing workflows with Kanban boards. Workflows in Jira make it easy to plan, track, release, and report on software.

How does it work in product management?

We noted above that product management is one of the common use cases for Jira Software. By integrating the tool with its product roadmap, a product team can also sync the status of its day-to-day tasks with its big-picture strategy.

Here are a couple of ways a Jira-roadmap integration can benefit your product team.

1. Pull Issue-Completion Updates from Jira into Your Roadmap

With the right product roadmap software, your product team can automatically sync updates of tasks in Jira with its roadmap. This will allow you to display an up-to-date picture of your development progress, and it will help make your team's roadmap a single source of truth.

2. Prioritize Issues from Jira Using Your Roadmap App

The right product roadmap software will also have tools to help you weigh the costs and benefits of initiatives and decide which your team should work on first. Here's how.

Let's say your team maintains a list of epics, stories, and other projects you'd like to work on, but you're not sure which to pursue next. You can import all of those items into your roadmap app, then use the app's Parking Lot platform to create side-by-side comparisons of the items and weigh them against each other according to factors such as:

- Customer value
- Implementation effort
- Operational costs
- Potential revenue
- Risks

How to Use JIRA?

Here is a step by step process on how to use Jira software:

Step 1: Open Jira software and navigate to the Jira Home icon

Step 2: Select Create project option

Step 3: Choose a template from the library

Step 4: Set up the columns as per your need from Board settings

Step 5: Create an issue

Step 6: Invite your Team members and start working

Planning your sprint

- Setting up your workspace
- Customizing your project
- Creating your backlog
- Grooming your backlog
- Planning your sprint
- Tracking your progress
- Wrapping up your work
- Doing work with your agile projects

A sprint is a short period (ideally two to four weeks) during which a development team implements and delivers a discrete product increment, e.g. a working milestone version. In this tutorial, your team will be working in two-week long sprints. Let's go ahead and create a sprint for your team.

Sprint planning:

Before creating and starting a sprint, your Scrum team would typically hold a sprint planning meeting. In this meeting, your team should:

- Review the estimates for selected issues
 - Break down the selected issues into an initial list of sprint tasks
 - Consider upcoming employee time-off, holidays, and other issues that may impact the completion of these sprint tasks
 - Gauge the team's capacity to complete these sprint tasks
- By the end of the meeting, your team should be confident enough to commit to completing the work in the sprint.
- In this tutorial, we will assume that the Teams in Space team can handle 20 story points of work in a sprint, and that everyone is

available for the full sprint. Typically, you would know how much work your team can complete in a sprint by reviewing information from past sprints, usually through velocity and sprint reports

Create a sprint

- On the Teams in Space board, click Backlog.
- Click Create Sprint at the top of the Backlog.
- Your new upcoming sprint will be added to your board, below any other future sprints.
- Select the Sprint 1 text and edit the name of the sprint to 'Spring Cleanup'.
- The top 4 issues in the Backlog are equal to 20 story points. This is what the team estimated that they could accomplish in the upcoming sprint.
- Drag and drop the top four issues from the Backlog into your new sprint.

The screenshot displays the Jira Backlog interface. At the top, there's a 'Backlog' header with a search bar and filters for 'Only My Issues' and 'Recently Updated'. Below this, there are two main sections for sprint backlogs. The first section, titled 'Spring Cleanup' (4 issues), has a 'Start Sprint' button and a 'Linked pages' link. It lists four issues: TIS-8 (4 points), TIS-2 (2 points), TIS-5 (8 points), and TIS-1 (6 points). The second section, titled 'Backlog' (4 issues), has a 'Create Sprint' button and lists four issues: TIS-3 (4 points), TIS-4 (3 points), TIS-6 (16 points), and TIS-7 (6 points). On the right side, there's a detailed view of the selected issue, TIS-1: 'Expand travel to destinations outside of The Solar System'. This view includes fields for 'Status' (TO DO), 'Component/s' (None), 'Labels' (None), 'Affects Version/s' (None), 'Fix Version/s' (None), 'Epic' (None), 'Reporter' (Abina [Administrator]), 'Assignee' (Unassigned), and 'Dates' (Created: 15/Nov/22 8.15 AM, Updated: 15/Nov/22 8.15 AM).

Start your sprint

Now that you have created a sprint, you can go ahead and start it.

- Click Start Sprint.
- Today's date and current time become the start date and time for the sprint. For the purpose of this tutorial, enter an end date of 5 minutes from the start date and time.
- Select Start to start the sprint and move the issues into Active sprints.

Creating a board

You need a board so that you can view and work on issues in JIRA Software. A board displays issues from one or more projects. You can either copy a board that someone else has created, or create a new board for yourself. You can create as many boards as you like.

There are two types of boards:

- Scrum boards are for teams that plan their work in sprints.
- Kanban boards are for teams that focus on managing and constraining their work-in-progress. Because work is not planned in advance using discrete time periods or versions, Kanban boards do not have a Backlog screen.

Before you begin:

Any user can create a board, but certain permissions are required to share the board with other users:

- If you create a board via Boards (in header) > View All Boards, you will not be able to share it, unless you have the 'Create Shared Objects' global permission.
- If you create a board via the following methods, you do not need the 'Create Shared Objects' global permission to share the board:
 - Creating a project (where a board is created for the project by default)
 - Setting up JIRA Software for the first time (where you're prompted to create a project, which also creates a board for the project)
 - Copying a board (the copied board will be shared with the same users as the original board)

Creating a new board

- Log in to JIRA Software.
- If you're setting up JIRA Software for the first time, do either of the following actions. Otherwise, please continue.

If you're creating either a Scrum Software Development project or a Kanban Software Development project, follow the prompts to create your project. A board is created for your project by default.

If you're creating a Basic Software Development project, follow the prompts to create your project. After your project is created, click Create board under your project name, and follow the prompts to create your board.

- Select Boards > View All Boards from the top navigation bar.
- Click Create board at the top-right of the page, and choose whether to create a Scrum board or Kanban board, as described below. Note that you cannot change the board type after creation (that is, a Scrum board cannot become a Kanban board, and vice versa).

To create a Scrum board based on projects

- Click Create a Scrum board, then choose whether to base your board on a new Software project ('Browse Projects' permission required) or an existing project. Note, you can also choose to Create a Scrum board with sample data ('Browse Projects' permission required). This will base your board on a new project pre-populated with sample data.
- Follow the prompts to set up your board and project (if you are basing your board on a new Software project). This will create a pre-configured Scrum board containing all the issues in your chosen project(s). Your Scrum board will have an issue filter with the following query:
 - `project = "[YOUR PROJECT(S)]" ORDER BY Rank ASC`

To create a Kanban board based on projects

- Click Create a Kanban board, then choose whether to base your board on a new Software project ('Browse Projects' permission required) or an existing project. Note, you can also choose to Create a Kanban board with sample data ('Browse Projects' permission required). This will base you on a new project pre-populated with sample data.
- Follow the prompts to set up your board and project (if you are basing your board on a new Software project). This will create a pre-configured Kanban board containing all the issues in your chosen project(s) that do not belong to a released version. Your Kanban board will have an issue filter with the following query:

```
project = "[YOUR PROJECT(S)]" AND (fixVersion in  
unreleasedVersions() OR fixVersion is EMPTY) ORDER BY Rank  
ASC
```

To create a Scrum or Kanban board that is based on a filter

Before you begin: You must have access to at least one saved issue filter (either your own filter, or one that someone else has shared with you). If you don't, first create and save a new issue filter (see the documentation on issue filters and JQL if you need help).

Click Create a Kanban board and base your board on an 'existing Saved Filter'. Follow the prompts to set up your board. Your new board will be available to all users who have access to your chosen filter. See the examples below for sample JQL for your board's filter.

Note, if you are the owner of the issue filter, you can edit it via the Edit Filter link. Otherwise, this link will not appear.

Your new board will be shown. At the top is a link that you may want to send to other people so that they can use your board.

Creating your backlog

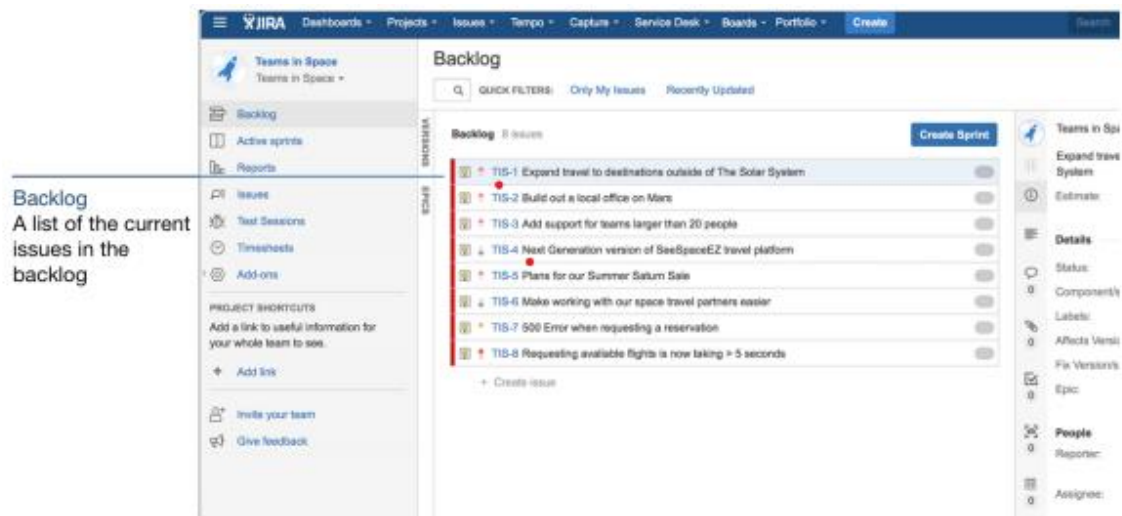
- Setting up your workspace
- Customizing your project
- Creating your backlog
- Grooming your backlog
- Planning your sprint
- Tracking your progress
- Wrapping up your work

➤ Doing more with your agile projects

Your backlog is a list of tasks that represents outstanding work in a project. Usually, a project would have issues in the backlog, and you can add these issues to a sprint so your team can work on them. Since Teams in Space is a new project, you won't have issues on your backlog. Let's create some work for your team.

Backlog

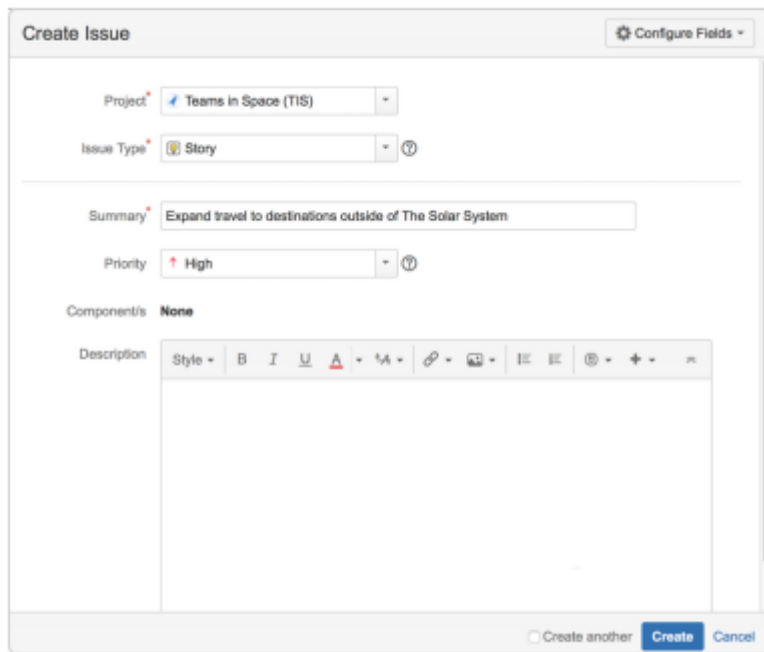
The Backlog gives you a place to organize your sprints. You can create new issues or sub-tasks, organize your backlog, create versions, organize via epics, and start sprints.



Create issues

By default, any team member can create issues. In this tutorial, however, you will create all of the backlog issues.

- On the Teams in Space board, choose Create at the top of the screen.



The screenshot shows the 'Create Issue' dialog box. The 'Project' field is set to 'Teams in Space (TIS)'. The 'Issue Type' field is set to 'Story'. The 'Summary' field contains the text 'Expand travel to destinations outside of The Solar System'. The 'Priority' field is set to 'High'. The 'Component/s' field is set to 'None'. The 'Description' field is a rich text editor with a toolbar. At the bottom, there are buttons for 'Create another', 'Create', and 'Cancel'.

Fill in the fields using the data shown below. Only the fields with * are mandatory.

- **Project:** Teams in Space Issue
- **Type:** Story
- **Summary:** Expand travel to destinations outside of The Solar System
- **Priority:** High
- Leave all other fields blank or at their default values.

- Choose Create to make a new issue. An issue key (TIS-1) is created for this issue, which comes in handy when searching for issues later.

Estimate in story points

- Plan for the team
- Customize the team board
- Estimate in story points
- Analyze team reports
- Optimize future plans

There's a huge variety of ways to estimate stories, but the objective of every approach is to be able to get better at predicting how much work the team can do each sprint. In agile scrum, this translates to knowing the team's velocity.

Velocity measures the number of 'estimation units' that a team usually completes from sprint to sprint. It is effectively a productivity rate based on an estimation of volume of work, and it is best worked out in a measure other than 'time'.

➤ What are story points?

Story points are a commonly used measure for estimating the size of an issue in scrum teams. During a typical planning session, a trivial bug fix might be estimated as a 1 or 2, and a larger feature might be anything up to a 12. Note that the scale of points does vary between scrum teams. Some use 1-12, others 1-5. The key is to use the same scale so that your velocity is consistently calculated.

If issues are estimated larger than this, they might need to be broken into smaller stories or subtasks.

The focus is on story points here because it's the more common scrum estimation method, and we use it at Atlassian. Your company may use hours or ideal days. If you do use time for estimation, you might want to have a look at [Configuring estimation and tracking](#)

Q: Why use story points instead of hours?

A: It is much more useful in the long term to roughly predict how much work your team can do each sprint, than it is to try to estimate how long each story will take in time.

Story points enable the team to estimate stories in comparison to other stories, instead of forcing them to determine the time it will take to complete each story. Velocity is then worked out based on how many points the team can complete in each sprint. After a few sprints, team velocity stabilizes and estimation accuracy increases (or rather the same degree of inaccuracy is applied).

The problem with using hours to estimate is that the team is likely to estimate using the 'working hours' of the sprint (with some buffer time) and this is almost never an accurate reflection of a story's complexity or size. Accurate velocity is difficult to establish when stories are estimated in isolation, instead of by comparison.

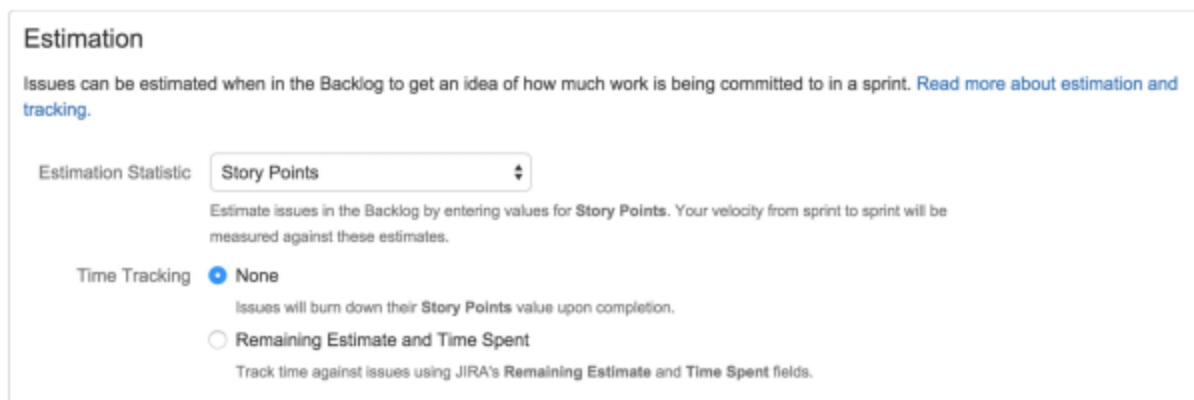
Estimation, time and velocity are really critical things to understand. Read more in the [Estimating an issue](#) topic.

Estimate in points, track in time :

Even when you estimate in story points, you can still track in time if you want. Knowing the team velocity (regardless of the measure used) enables you to roughly guess how long estimated backlog items will take to complete.

But JIRA Software does also have a couple of dedicated fields (Remaining Estimate and Time Spent) to track time while using story points.

Go to Board > Configure > Estimation.



The screenshot shows the 'Estimation' configuration page in JIRA. At the top, it says 'Estimation' and provides a brief explanation: 'Issues can be estimated when in the Backlog to get an idea of how much work is being committed to in a sprint. [Read more about estimation and tracking.](#)'. Below this, there is a section for 'Estimation Statistic' with a dropdown menu currently set to 'Story Points'. A note below the dropdown states: 'Estimate issues in the Backlog by entering values for Story Points. Your velocity from sprint to sprint will be measured against these estimates.' Underneath, there is a 'Time Tracking' section with two radio button options. The first option, 'None', is selected and has a sub-note: 'Issues will burn down their Story Points value upon completion.' The second option is 'Remaining Estimate and Time Spent', which is currently unselected and has a sub-note: 'Track time against issues using JIRA's Remaining Estimate and Time Spent fields.'

- Select the Estimation Statistic (unit of estimation) - choose from story points, original time estimate, and issue count.
- Switch on the Remaining Estimate and Time Spent option to get a more accurate picture of how things are tracking in time units.
- If you leave Time Tracking as None, you can still refer to reports such as the Burndown chart to monitor progress

Find out more about how time tracking work in projects in the configuring estimation and tracking topic

Fun fact: Inaccuracy is good

The goal of velocity is to be able to look at a backlog of not particularly well-understood stories, and guess how many sprints it will take to complete. It requires a similar level of uncertainty for all of the estimates in the backlog. Determining accurate velocity relies on the equality of the inaccuracy, which is why you should NEVER re-estimate issues after a sprint has started.

Burndown Chart:



A Burndown Chart shows the actual and estimated amount of work to be done in a sprint. The horizontal x-axis in a Burndown Chart indicates time, and the vertical y-axis indicates cards (issues).

Use a Burndown Chart to track the total work remaining, and to project the likelihood of achieving the sprint goal. By tracking the remaining work throughout the iteration, a team can manage its progress, and respond to trends accordingly. For example, if the Burndown Chart shows that the team may not likely reach the sprint goal, then the team can take the necessary actions to stay on track.

Before you begin

The Burndown Chart only applies to Scrum boards.

Viewing the Burndown Chart

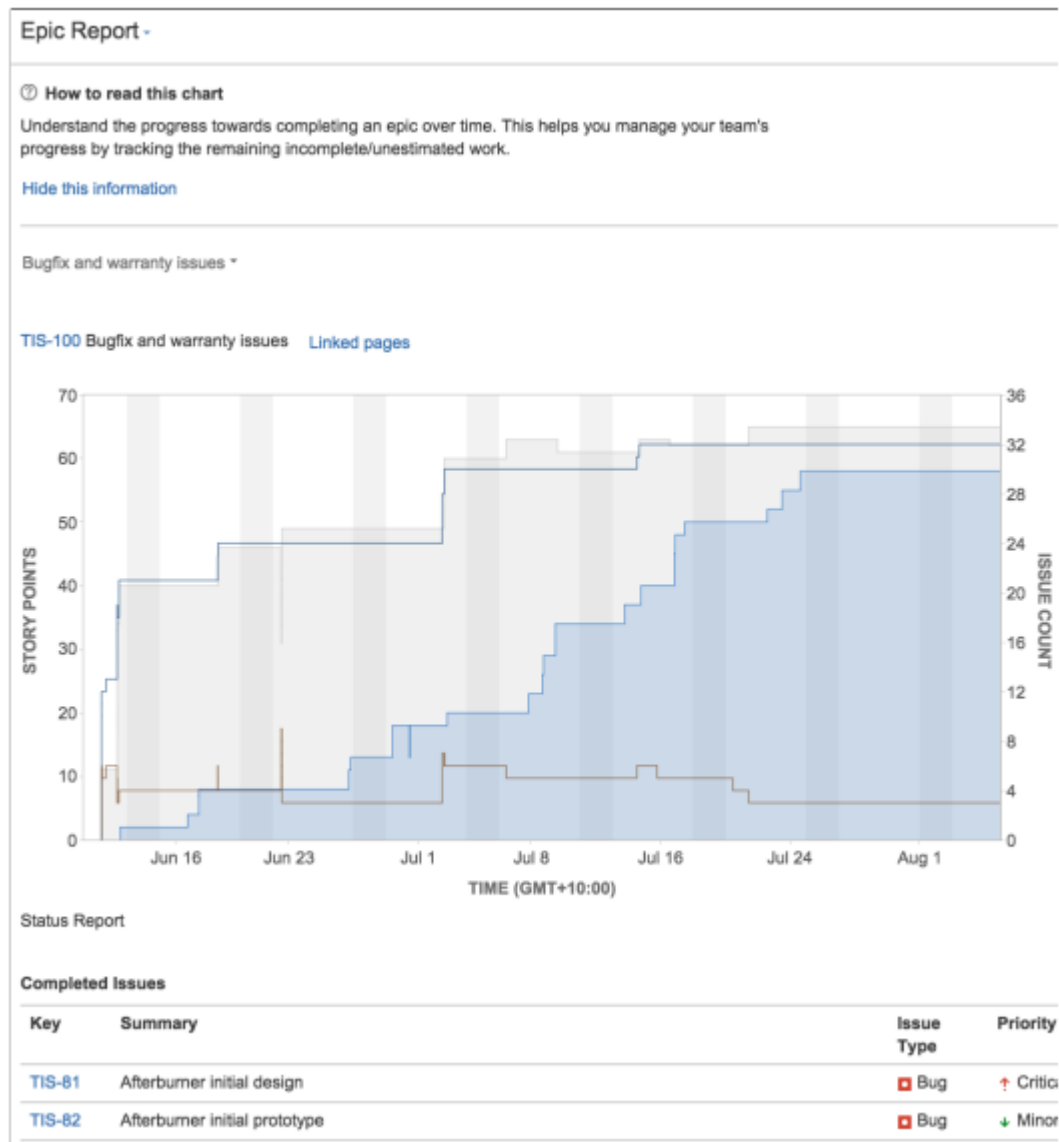
1. Click **Boards** (in header) > select your desired board
2. Click **Reports**, then select **Burndown Chart**.
 - To choose a different sprint, click the sprint drop-down.
 - To choose a different estimate statistic, click the estimation statistic drop-down. This change will be saved for you, for when you next visit this chart.

Understanding the Burndown Chart:

Before you start using the Burndown Chart, you should get to know how it works. The following information will help you understand the key functionalities of the Burndown Chart:

- The Burndown Chart is board-specific – that is, it will only include issues that match your board's saved filter.
- The vertical axis represents the estimation statistic that you have configured for your board.
- The Burndown Chart is based on your board's column mapping. An issue is considered to be 'To Do' when it is in a status that has been mapped to the left-most column of your board. Similarly, an issue is considered to be 'Done' when it is in a status that has been mapped to the right-most column of your board.
- If the grey 'Guideline' line does not show, the sprint may have been started before any issues were assigned to it.

Epic Report:



The Epic Report shows a list of complete, incomplete, and unestimated issues in an epic. It is particularly useful in planning work for an epic that may extend over multiple sprints.

Use the Epic Report to understand the progress towards completing an epic over time, and to track the amount of remaining work that's incomplete or unestimated.

Before you begin

This page only applies to Scrum boards.

Viewing the Epic Report

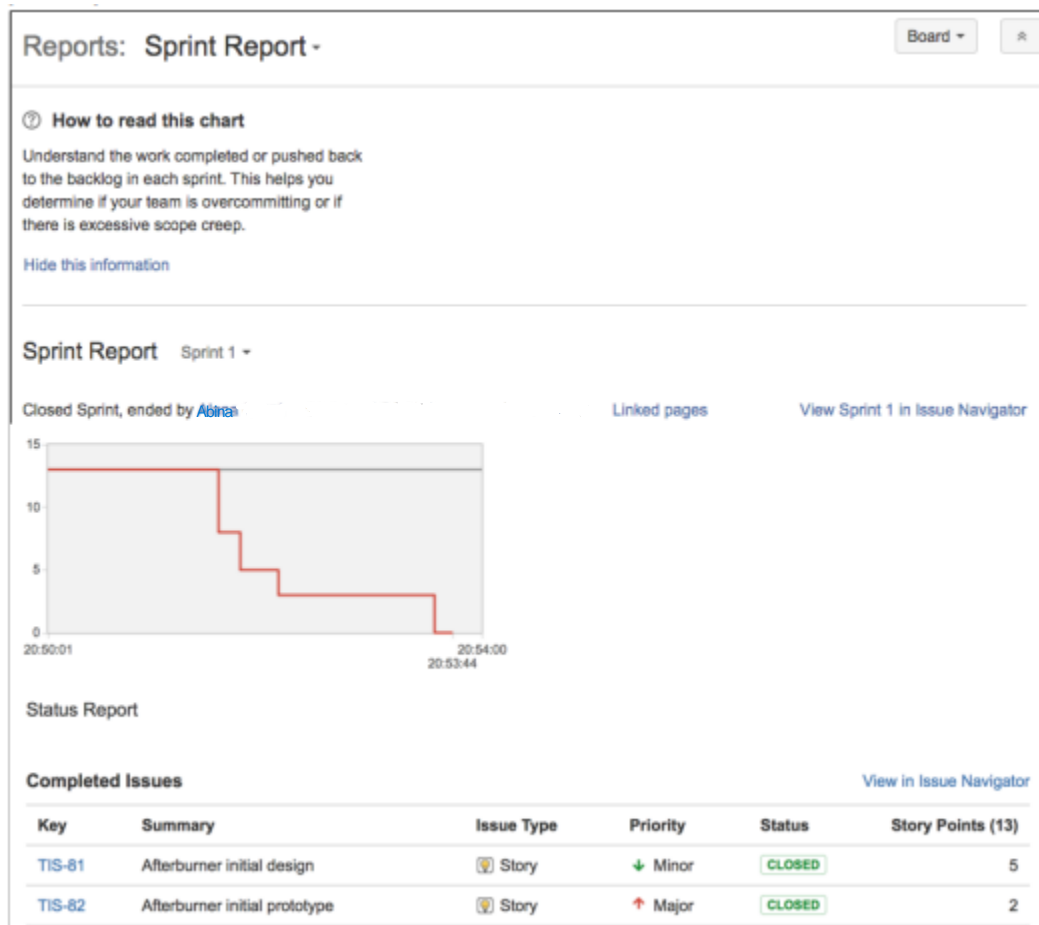
1. Click **Boards** (in header) > select your desired board.
2. Click **Reports**, then select **Epic Report**.
3. Select the relevant epic from the epic drop-down.
4. Click **View in Issue Navigator** to jump to the [Issue Navigator](#) and see a list of all the epic's issues.

Understanding the Epic Report

Before you start using the Epic Report, you should get to know how it works. The following information will help you understand the key functionalities of the Epic Report:

- The Epic Report is based on your board's column mapping. An issue is considered to be 'To Do' when it is in a status that has been mapped to the left-most column of your board. Similarly, an issues considered to be 'Done' when it is in a status that has been mapped to the right-most column of your board.
- The graph will look different if you are using Issue Count as your Estimation Statistic (rather than Story Points, as shown in the screenshot above).

Sprint Report:



The Sprint Report shows the list of issues in each sprint. It is useful for your Sprint Retrospective meetings, and also for mid-sprint progress checks.

Before you begin

This page only applies to Scrum boards.

Viewing the Sprint Report

1. Click **Boards** (in header) > select your desired board.
2. Click **Reports**, then select **Sprint Report**.
3. Select the relevant sprint from the sprint drop-down.

Understanding the Sprint Report

Before you start using the Sprint Report, you should get to know how it works. The following information will help you understand the key functionalities of the Sprint Report:

- The Sprint Report is board-specific – that is, it will only include issues that match your board's saved filter.
- Issues added after the sprint starts are indicated with an asterisk.
- An issue is considered to be 'To Do' when it is in a status that has been mapped to the left-most column of your board. Similarly, an issue is considered to be 'Done' when it is in a status that has been mapped to the right-most column of your board.

Velocity Chart:



The Velocity Chart shows the amount of value delivered in each sprint, enabling you to predict the amount of work the team can get done in future sprints. It is useful during your sprint planning meetings, to help you decide how much work you can feasibly commit to. You can estimate your team's velocity based on the total Estimate (for all completed stories) for each recent sprint. This isn't an exact science — looking at several sprints will help you get a feel for the trend. For each sprint, the Velocity Chart shows the sum of the Estimates for complete and incomplete stories. Estimates can be based on story points, business value, hours, issue count, or any numeric field of your choice.

The velocity can be estimated as the average, over several recent sprints, of the sum of the estimates for the amount of work completed by a team per sprint — so in the chart above, the velocity = $(13 + 10 + 17 + 26 + 23) / 5 = 17.8$. A team's recent velocity can be useful in helping to predict how much work can be completed by the team in a future sprint.

Viewing the Velocity Chart:

1. Click **Boards** (in header) > select your desired board.
2. Click **Reports**, then select **Velocity Chart**.
3. The Velocity Chart will be displayed, showing your last seven completed sprints

Understanding the Velocity Chart:

Before you start using the Velocity Chart, you should get to know how it works. The following information will help you understand the key functionalities of the Velocity Chart:

- The Velocity Chart is board-specific – that is, it will only include issues that match your board's saved filter.
- The Velocity Chart is based on your board's column mapping. An issue is considered to be 'To Do' when it is in a status that has been mapped to the left-most column of your board. Similarly, an issue is considered to be 'Done' when it is in a status that has been mapped to the right-most column of your board.

Purpose of a Jira project:

A Jira project is a collection of issues. Your team could use a Jira project to coordinate the development of a product, track a project, manage a help desk, and more, depending on your requirements. A Jira project can also be configured and customized to suit the needs of you and your team.

Conclusion:

Thus we conclude the report, the jira software is project management tool, used to manage the overall project outcome.