

The background features a light blue and white geometric pattern. Several hexagonal shapes are scattered across the page, some containing network diagrams with nodes and connecting lines. These hexagons are in various shades of orange and grey. Some hexagons are connected by thin grey lines, forming a larger network structure. A large, solid orange hexagon is positioned in the bottom left corner.

# WORKING IN JIRA

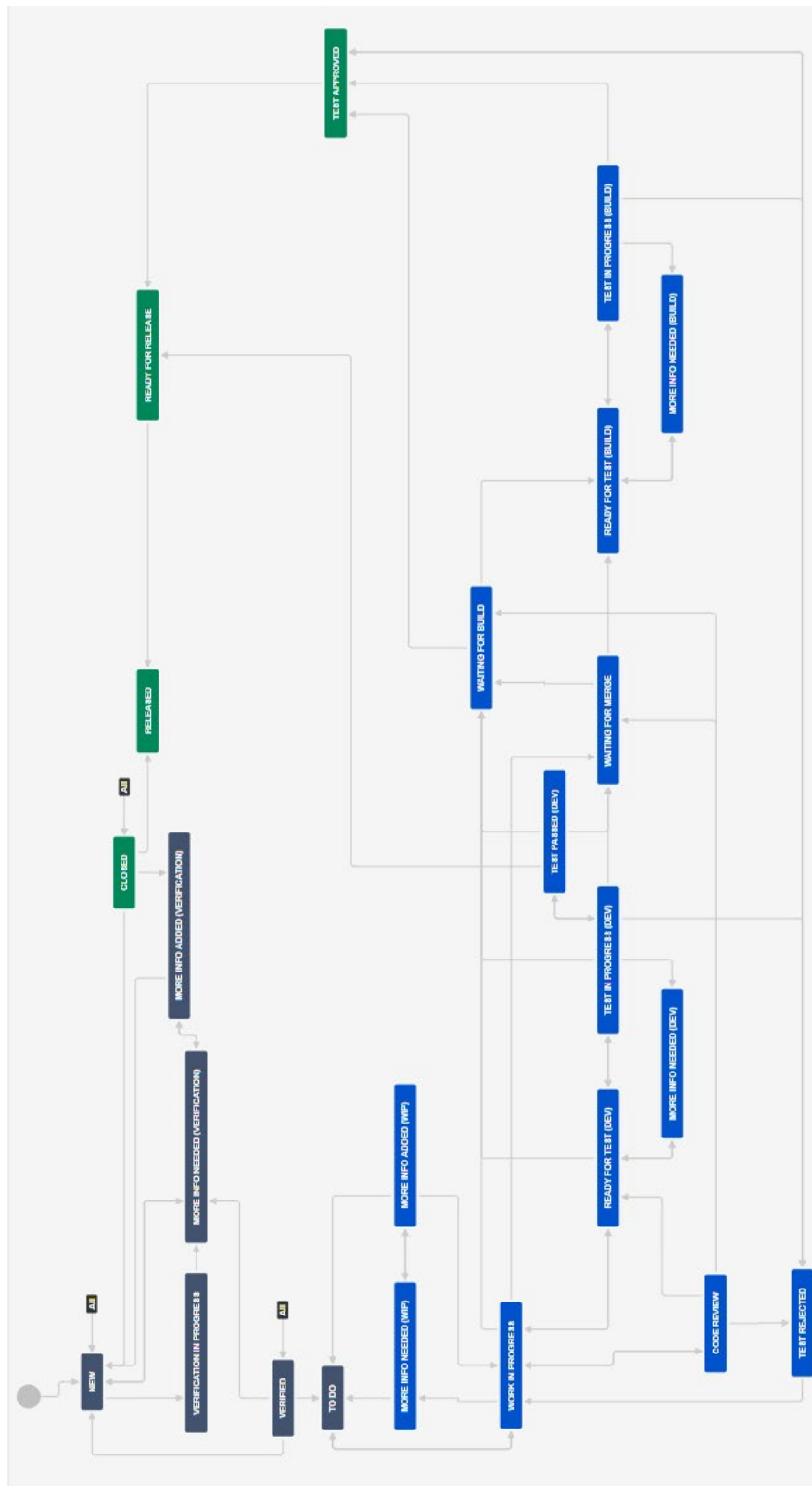
## Contents

|   |    |
|---|----|
| Contents .....  | 1  |
| Working in Jira .....   | 2  |
| Issue Workflow .....  | 4  |
| Recommendations.....  | 6  |
| Configura Bot.....  | 6  |
| Working with Issues Escalated from Jira Service Desk .....              | 7  |
| Issue / Ticket Status Explanation .....                                 | 10 |
| Epic Status Explanation .....   | 11 |
| Issue / Ticket Resolution Explanation.....                              | 12 |
| Dashboards .....  | 13 |
| Backlog View.....   | 18 |
| Board (Kanban) / Active Sprint (Scrum) View.....                        | 19 |
| Scrum vs Kanban.....  | 20 |
| Changing Issue Status in the Board/Active Sprint View .....             | 21 |
| Releases View .....   | 22 |
| Reports View .....  | 25 |
| Issues View .....   | 26 |
| Creating a new Issue.....   | 30 |
| Moving an Issue.....  | 32 |
| Editing an Issue.....   | 33 |
| Searching for Issues .....  | 34 |
| Components View .....   | 35 |
| Submitting issues from CET Designer directly to partner's project ..... | 36 |
| Appendix 1: Global Project Automation Rules .....                       | 37 |
| Jira Software Project.....  | 37 |
| Jira Service Desk .....   | 39 |
| Appendix 2: Jira Software Keyboard Shortcuts .....                      | 40 |

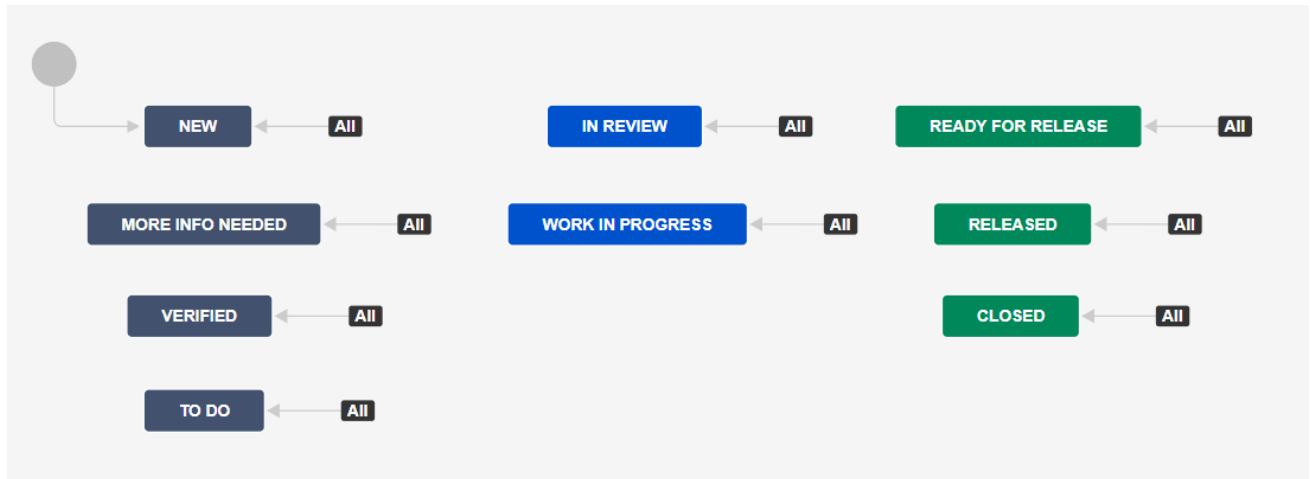
## Working in Jira

Below is a graphical overview of the workflow for working with issues in Jira. An up-to-date version of this can always be accessed by clicking the **View workflow** link next to the issue status in Jira.

Graphical overview of all issue types except Epic:



Graphical overview of Epics:



## Issue Workflow

1. *New* issues are created.
2. Set the issues to *Verified* if you have all the necessary info.
3. Product Owner to go through the new issues and prioritize the issues in the backlog view.
4. Scrum master or Product owner can group the issues under the same epic depending on the product line or project phase. Epic status can be used to indicate different project stages. (optional)
5. Scrum master or Product owner to move the issues to different sprint versions. They can either drag issues from the backlog to a sprint OR right-click on issues in the backlog and choose to send to different sprint versions.
  - Create different quick filters in the backlog (for example for different priorities, newly created issues etc.)
  - Create special “pre-sprint” backlog for higher priority issues.
  - Set the epics.
  - Note: a sub-task is always assigned to the same sprint as its parent task.
6. Developers or Scrum master to estimate the issues. Use either story point or original time estimates. Set issues to *To Do*. Issues can be estimated in the upcoming sprint and/or the special backlog. Note: developers should log their work in terms of time spent and remaining time, once they begin working.
7. When a developer or QA or product owner needs more information to verify an issue, resume work or continue testing, add a comment and set the issues to *More info needed (Verification)*, *More info needed (Work in progress)*, *More info needed (Dev)*, or *More info needed (Build)*. Make sure the assignee is set correctly.
8. An assignee of a *More info needed* issue can answer a question by adding a comment and set the issue to *More info added (Verification)*, *More info added (Work in progress)* or *Ready for Test (Dev)* or *Ready for Test (Build)*. Make sure the assignee is set correctly.
9. Developers use the scrum board to work in the sprint. Please note that moving a card up or down in a column will change its rank.
10. Developers can either drag cards from one column to another to update the issue status or use the Transition button on the View Issue screen. When dragging a card to other columns, built-in rules will provide visual feedback on where the card may be placed.
11. The *Work in Progress* status should be used when a developer starts working on the issue.
12. Before setting the status to *Ready for Test (Dev)* or *Ready for Test (Build)*, please fill out the following:
  - Description: any relevant background information and test instructions.
  - Fixed in version: feature branch or Extension version.

13. File issues during testing:
  - If the issue is straight-forward/one test case, test reject without filing any new issue.
  - If the issues have more test cases/test scenarios, file issue as separate issues and link those issues to the parent test ticket.
14. Add test result and update the status (*Test rejected* or *Waiting for build*). Test result should show the failed test cases and special notes for developer. You can add the issue number in the test result where it will become a hyperlink to that issue.
15. Sprint review and Retrospective. The Scrum master can use the reporting tool in JIRA to track and analyze the team's work throughout a project.
16. Sprint Retrospective. Create the Retrospective document in your own SharePoint.
17. After QA set an issue to *Waiting for Build*, the developer can set it to *Ready for Test (Build)* for the customer to test in the build. (optional)
18. Developer can set *Test Approved* issued to *Ready for Release* if all testing is done.
19. QA: s to run regression test and/or smoke test.
20. When the build is released, set the Release version to Released. All issues that are in that version and have status *Ready for release* will be transitioned to *Released*.
  - If you transition an issue from *Ready for release* to *Released* (either manually or by using the above feature), it will send an email to the reporter about it.
21. QA to update the regression test cases with the new changes.

*For more information on how to set up boards, please see*

[https://downloads.configura.com/marketing/resources/Documents\\_Jira.pdf](https://downloads.configura.com/marketing/resources/Documents_Jira.pdf)

## Recommendations

- Work in the Board/Active Sprint and Backlog views. They are highly visual and provide great visual support on what is allowed, how things may move etc.
- Jira has very powerful and flexible filters that are publicly shareable and work across multiple projects.
- Use Quick Filters.
- Create a special “pre-sprint” backlog for higher priority tickets.
- Set the epics.
- Use the Bulk Change to edit multiple issues simultaneously.
- Use mentions (@name) to ask a question or share information with a specific user within Jira.

## Configura Bot

The Configura Bot will automatically perform certain actions following an action or a status transition. Should it not be setup correctly, however, you can edit the field manually.

For example: When an issue is transitioned to *Work in Progress*, the assignee will automatically be set to *Developer* if the developer field is set. If the developer field is empty, the assignee will automatically be set to current user. For more information, see Appendix 1.

## Working with Issues Escalated from Jira Service Desk

If your Jira Software Project includes issues that are escalated from Jira Service Desk, there are some things that are good to know. If your organization does not have that connection, you may skip this section.

When an issue is escalated from a Service Desk project, a linked issue is created in the Software project. Summary, Description, Troubleshooting steps, Additional and Attachments are copied to the linked issue. In Troubleshooting steps, the support agent can choose to specify which steps have been taken by support before being escalated. The Troubleshooting steps are displayed as a Comment in the Activity section.

Comments are *not* copied, but the developer can use the link to go to the Service Desk issue and view the comments:

FD-52022 [Return to queue](#)

### Test Emil

[Edit](#) [Comment](#) [Log work](#) [More](#) [Email](#) [Share](#) [Export](#)

- Details
- Description
- Attachments
- Issue Links
- Activity

People

Dates

Agile

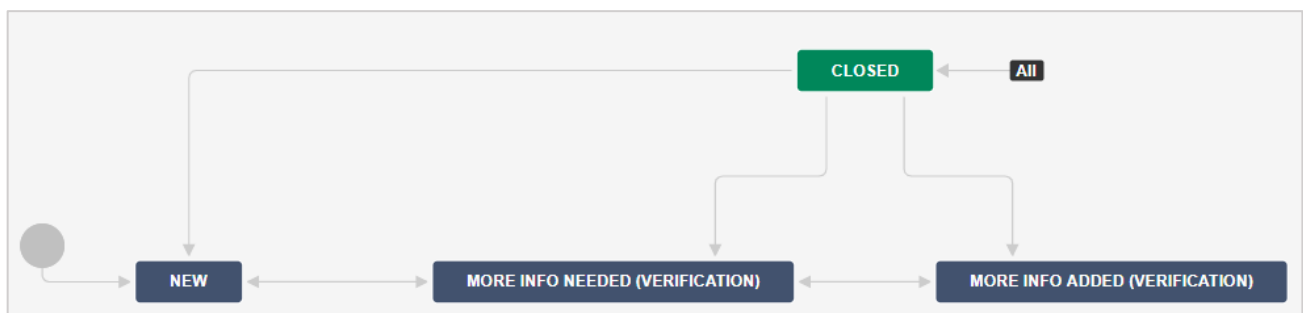
Issue Links

escalated from

FS-79 Test Emil [ESCALATED](#)

[Comment](#)

Below is a graphical overview of the workflow for **User Bug Reports** and **User Feature Requests**:





If the developer wishes to share information or ask a question, they can use @support in a comment. The comment is then added to the Service Desk issue and the issue status changes to *Waiting for support*. The status of the Software issue will automatically be changed to *More info needed*.

Vice versa, support agents can use @developer to add a comment to the Software issue. The Configura Bot will update the Jira Software issue status automatically depending on the previous issue status. For more information, see Appendix 1.

The developer can request additional files, for example log files or drawings from the Customer/Reporter through the support agent. When the Customer/Reporter provides additional files, they become available only on the Service Desk issue. The support agent can then share a comment with the developer and write "See linked issue for additional files".

A developer can access the Service Desk issue without a Jira Service Desk license, but they cannot edit the ticket except make internal comments or attach files.

When a Service desk issue is resolved and closed, there will be an automatic comment added to the Software project that the Service Desk issue has been resolved:

---

▼  Configura Bot added a comment - Just now

Support issue got resolution:  
Done

---

Please note that User Bug Reports and User Feature Requests always come in through support, and it is the responsibility of the Product Owner to review them, decide if action should be taken, and, in that case, convert them to regular issues. For more information, see the *Issue Type* and *Moving an issue* sections.

When a Software issue with a Service Desk link is set to *Released* by the developer, an automatic comment is shared with the Customer/Reporter.

Edit Comment Assign More Release Closed

Details

Type:

☒ Task

Status:

READY FOR RELEASE [\(View Workflow\)](#)

Priority:

High

Resolution:

Done

Affects Version/s:

None

Fix Version/s:

Pilot

Labels:

None

When an entire version is set to *Released* in Software, the status of each individual issue is also set to *Released*, meaning that each of the linked Service Desk issues will share automatic comments with the Customer/Reporter the same way as described above.

Releases

Manage Versions

QUICK FILTERS: Released Unreleased Archived

Merge versions

Add

| Version              | Status                  | Progress               | Start date | Release date | Description | Actions          |
|----------------------|-------------------------|------------------------|------------|--------------|-------------|------------------|
| <span>Phase 2</span> | <span>UNRELEASED</span> | No issues              | 2019-10-07 | 2020-04-30   |             | <span>...</span> |
| <span>Phase 1</span> | <span>UNRELEASED</span> | No issues              | 2019-06-03 | 2020-01-01   |             | <span>...</span> |
| <span>Pilot</span>   | <span>UNRELEASED</span> | <div><div></div></div> | 2019-01-21 | 2019-08-15   |             | <span>...</span> |

Atlassian Jira Project Management Software (v8.1.0#801000-sha1:2e1cd1b) · [About Jira](#) · [Report a problem](#)

ATLASSIAN

Release  
Build and Release  
Archive  
Delete  
Edit

## Issue / Ticket Status Explanation

| STATUS                              | Notes:   |
|-------------------------------------|--|
| NEW                                 | All newly created tickets should be set to new.  |
| VERIFICATION IN PROGRESS            | Person in charge is verifying an issue.  |
| VERIFIED                            | Person in charge is responsible for checking and setting the tickets to verified, which can be done when they have all the necessary info. |
| MORE INFO NEEDED (VERIFICATION)     | Person in charge needs more info before setting the status to verified.  |
| MORE INFO ADDED (VERIFICATION)      | More info added back to the questioner so that the ticket can be verified.   |
| TO DO                               | Developer has estimated the ticket.  |
| WORK IN PROGRESS                    | Developer is working on the ticket (fixing or implementing new features).  |
| MORE INFO NEEDED (WORK IN PROGRESS) | Developer needs more info on how to fix the issue or implement the changes.  |
| MORE INFO ADDED (WORK IN PROGRESS)  | More info added for the questioner so that the work can resume.  |
| CODE REVIEW                         | Code review for developer.   |
| READY FOR TEST (DEV)                | Ready for testing in developer version.  |
| TEST IN PROGRESS (DEV)              | QA is testing the ticket in developer version.   |
| MORE INFO NEEDED (DEV)              | QA needs more information from developers to proceed with testing in developer version.  |
| TEST PASSED (DEV)                   | Testing is approved in developer version. Merging is not needed for now.   |
| WAITING FOR MERGE                   | Testing is approved in developer version, and it is ready for merge.   |
| WAITING FOR BUILD                   | Merge request is merged and ready for build.   |
| READY FOR TEST (BUILD)              | Ready for testing in build version.  |
| TEST IN PROGRESS (BUILD)            | QA is testing the ticket in build version.   |
| MORE INFO NEEDED (BUILD)            | QA needs more information from developers to proceed with testing in build version.  |
| TEST REJECTED                       | The ticket has not passed testing and is rejected.   |
| TEST APPROVED                       | The ticket has passed testing and is approved.   |
| READY FOR RELEASE                   | Fixes are added in the build, but the build is not yet released.   |
| RELEASED                            | The fixes or new features are released.  |
| CLOSED                              | Closed the issue with different resolution.  |

## Epic Status Explanation

| STATUS            | Notes:  |
|-------------------|---|
| NEW               | All newly created epics should be set to new.   |
| VERIFIED          | Person in charge is responsible for checking and setting the epics to verified, which can be done when they have all the basic information to proceed to planning stage. High level of estimation can be done during this stage (optional). |
| MORE INFO NEEDED  | Person in charge needs more information before setting the epic status to verified.   |
| IN REVIEW         | This is a planning stage. Pre-study, adding more information, estimation and prioritization should take place at this stage.  |
| TO DO             | Developer has estimated the ticket.   |
| WORK IN PROGRESS  | Developer is working on the issues of this epic (fixing or implementing new features).  |
| READY FOR RELEASE | All issues of this epic are added in the build, but the build is not yet released.  |
| RELEASED          | All issues of this epic are released.   |
| CLOSED            | Closed the epic with different resolution.  |

## Issue / Ticket Resolution Explanation

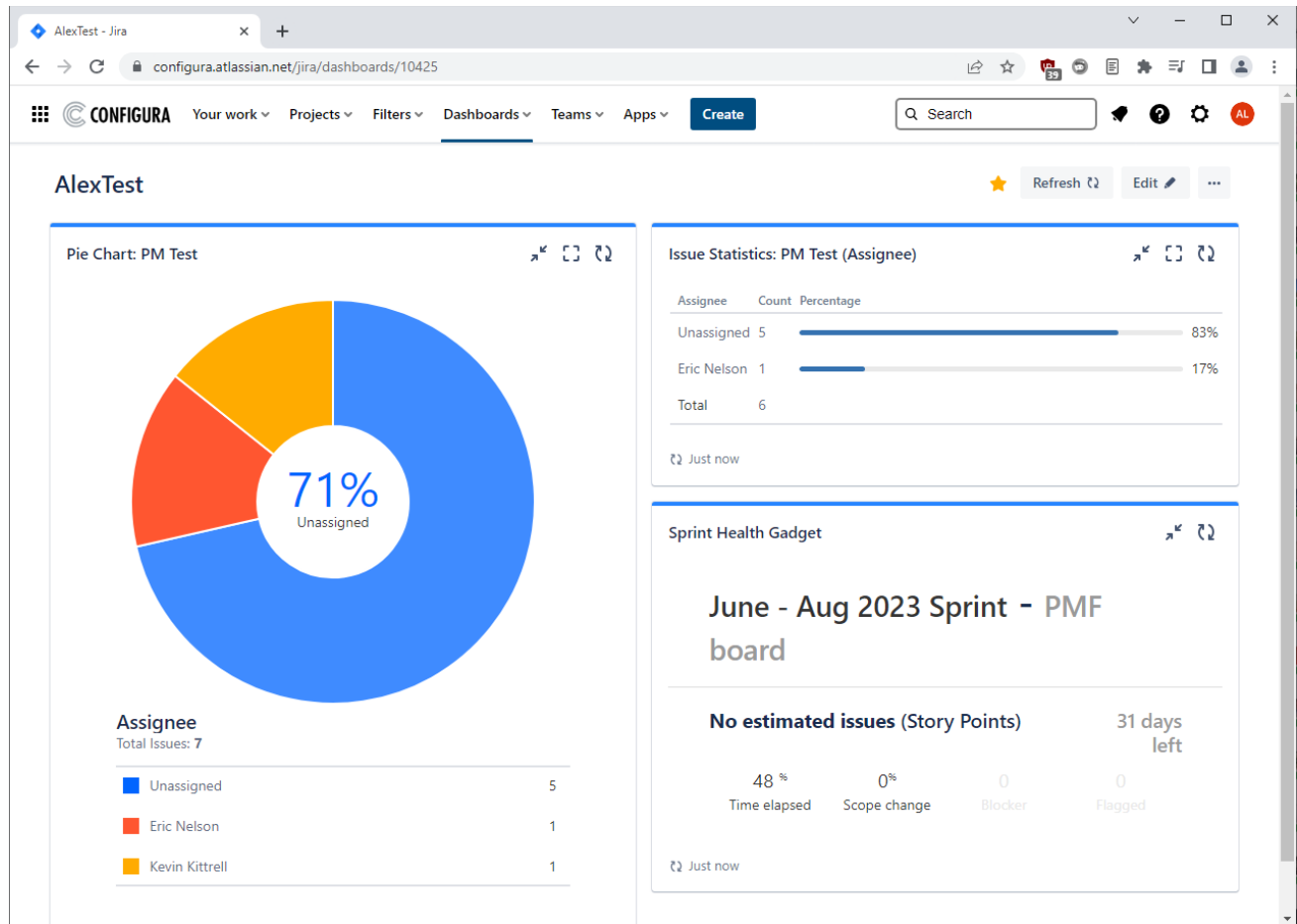
These are the different resolutions that can be used when closing an issue.

| RESOLUTION             | Notes:   |
|------------------------|--|
| Done                   | Work has been completed on this issue.   |
| Duplicate              | The problem is a duplicate of an existing issue (Existing issues should be linked to the duplicate.)   |
| Cannot reproduce       | This issue cannot be reproduced in the latest version.   |
| Declined               | This issue was rejected.   |
| Hardware failure       | This issue was caused by physical items connected to your computer, such as keyboard, mouse, RAM, or hard drive.   |
| Software failure       | This issue was caused by third party software, such as drivers, Windows, antivirus or any 3 <sup>rd</sup> party program causing havoc.                     |
| Revisit                | This issue is a verified bug, feature, or task but is currently of low priority. Product owner should revisit these issues on a regular basis to reassess. |
| No response            | There was no response from user after a request for more information.  |
| Update / Clean restart | An update or a clean restart solved the problem.   |

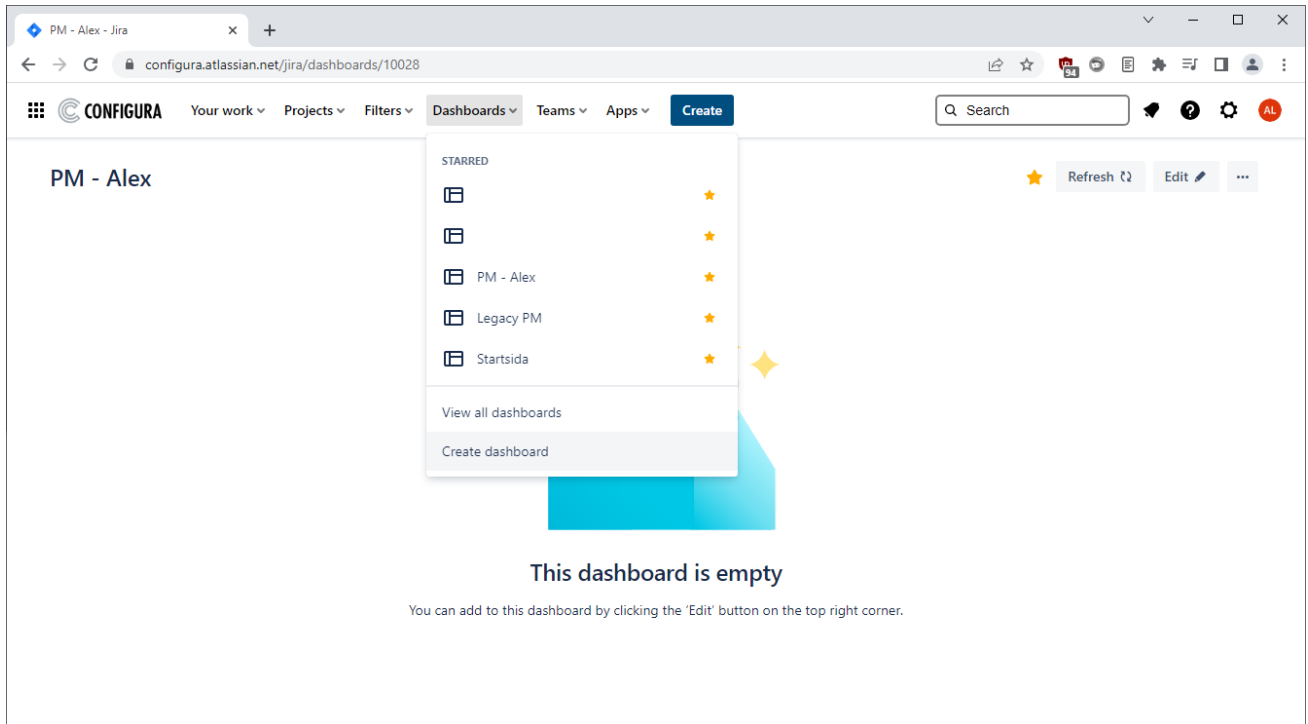
## Dashboards

Logging on to Jira displays the **Dashboard**, which is intended to give you a high-level view of the project quickly. Multiple dashboards are supported, and they each consist of a collection of *gadgets* that you can customize and reorganize to your own needs.

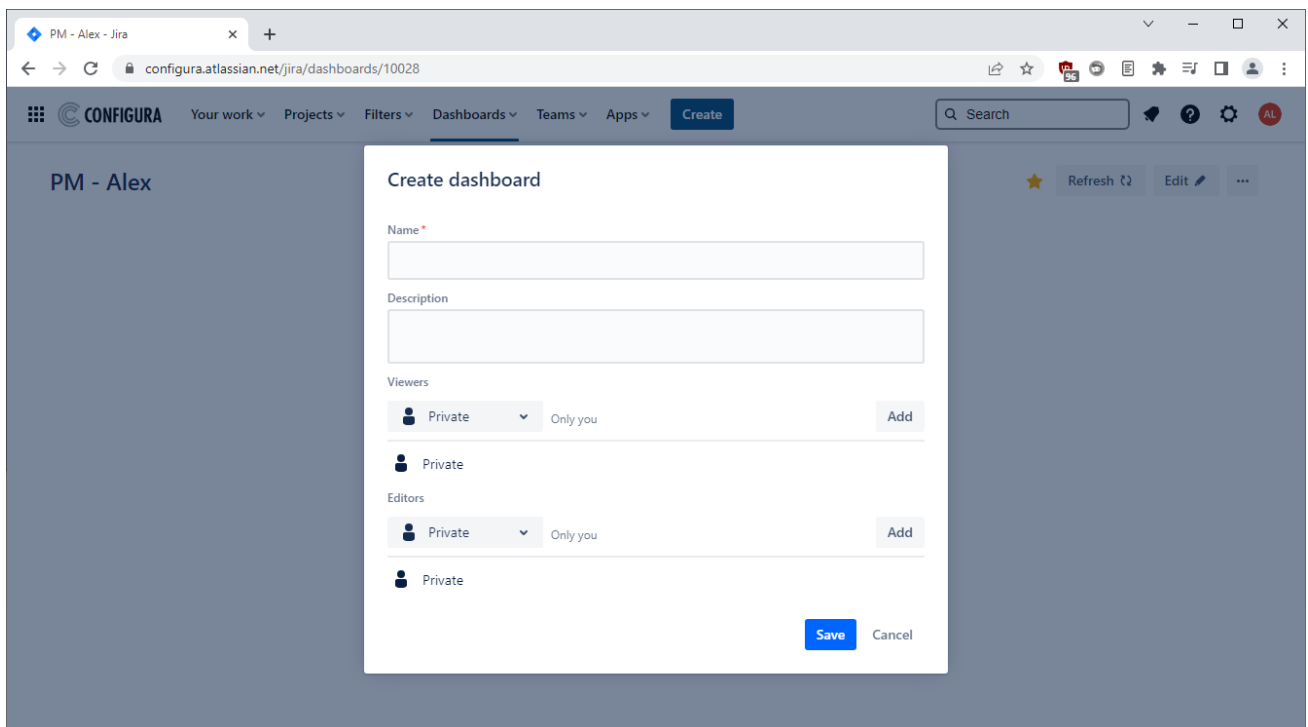
### Example



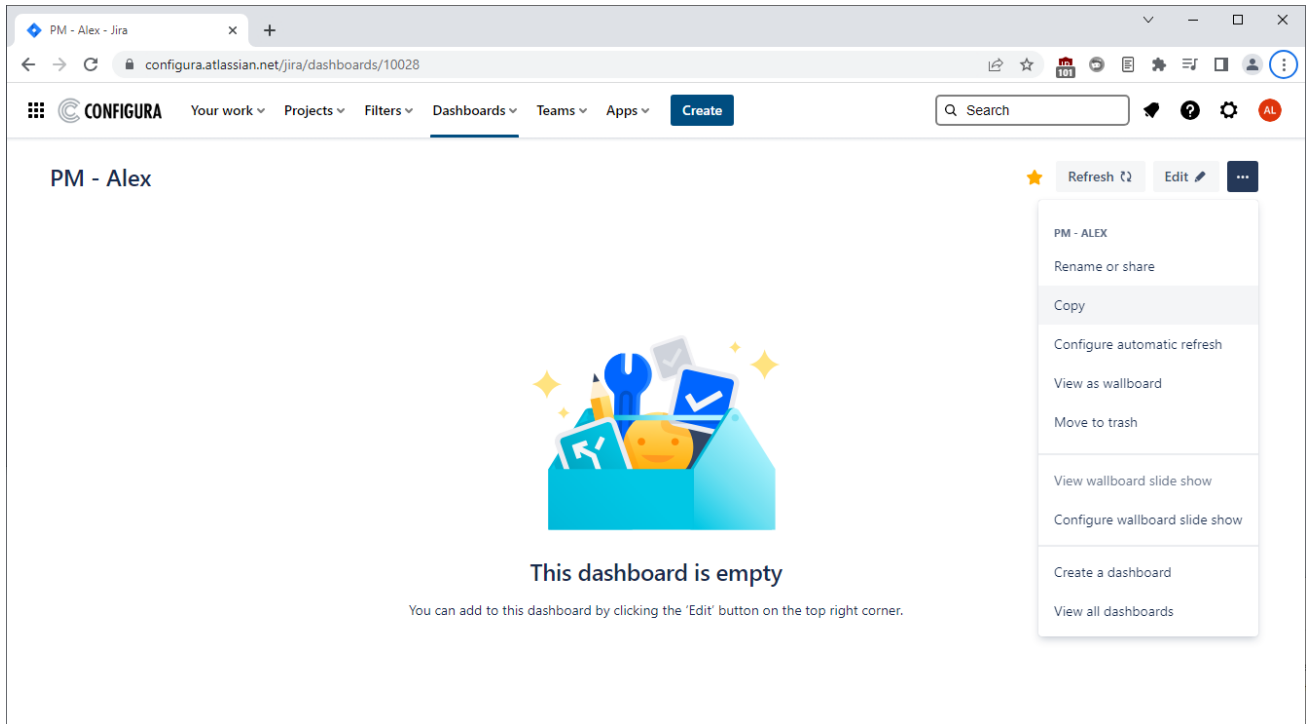
To create a new Dashboard, go to the Dashboard menu and Click the Create dashboard button:



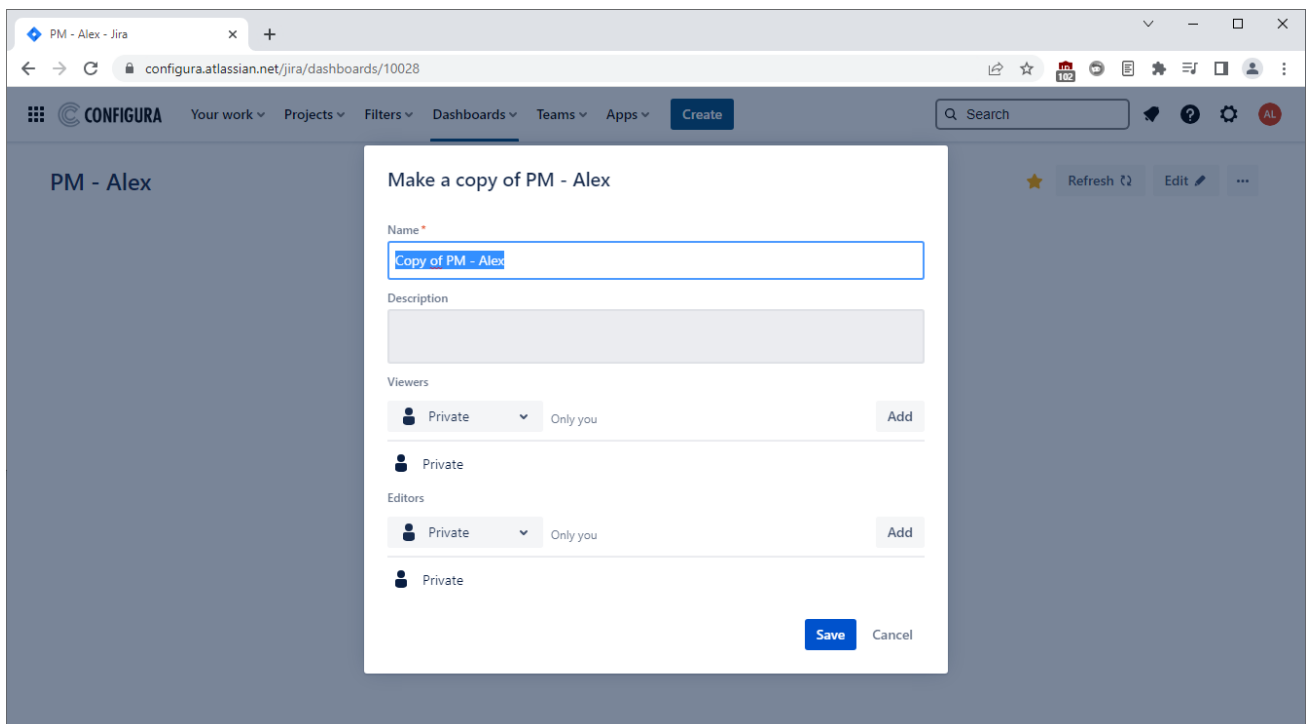
Enter a name and optionally choose viewers and editors. When done, click Save:



To create a Dashboard from an existing one, open the Dashboard, open the dotted menu and click Copy:

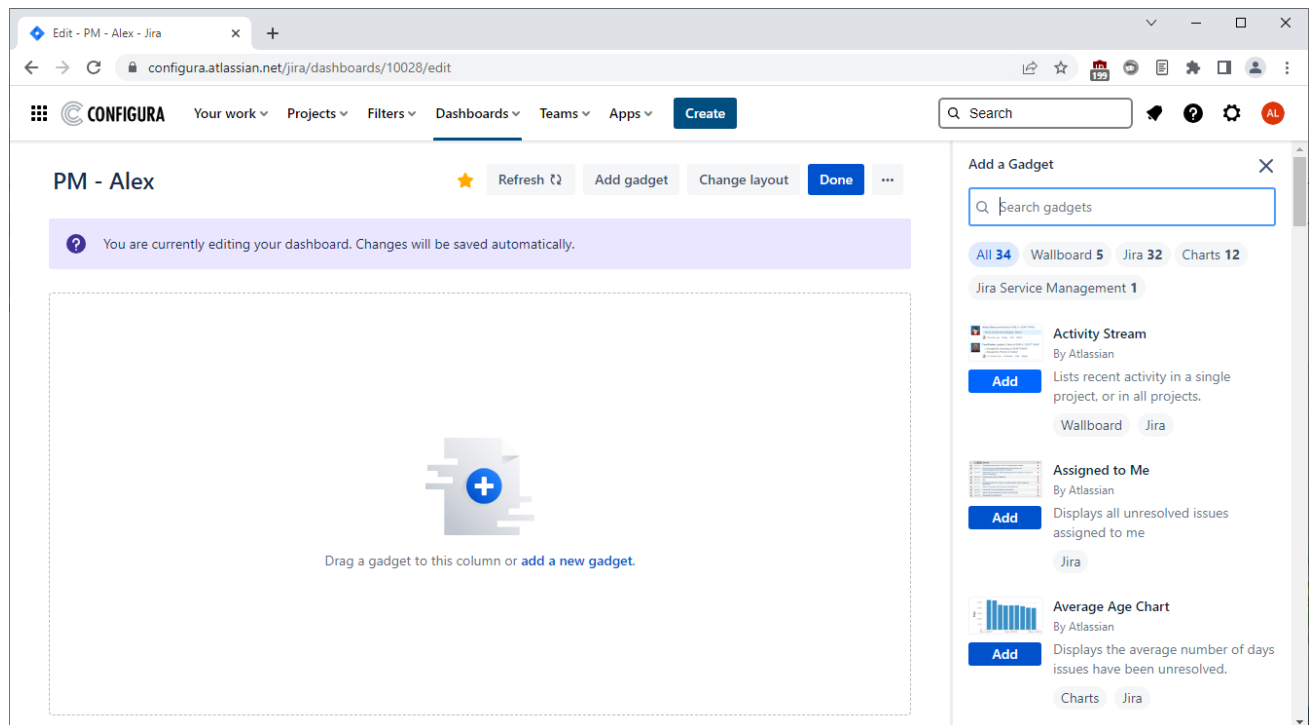


Enter a name and optionally choose viewers and editors. When done, click Save:

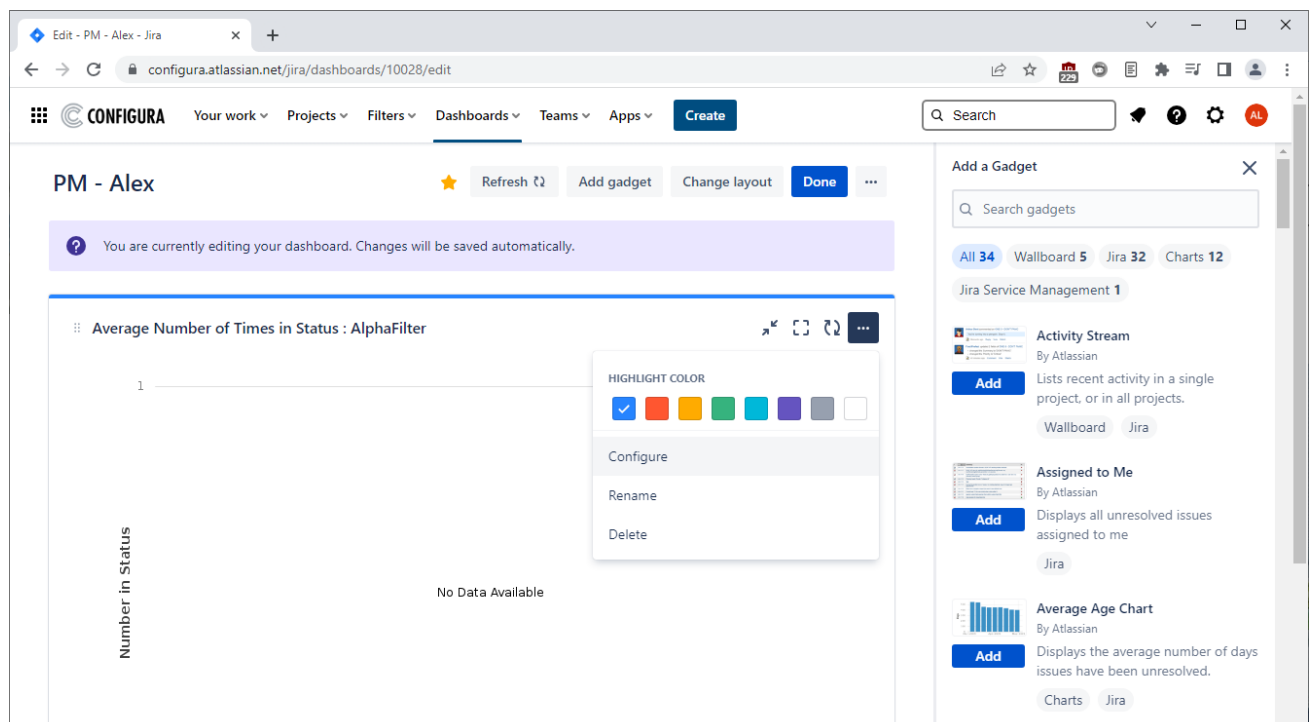




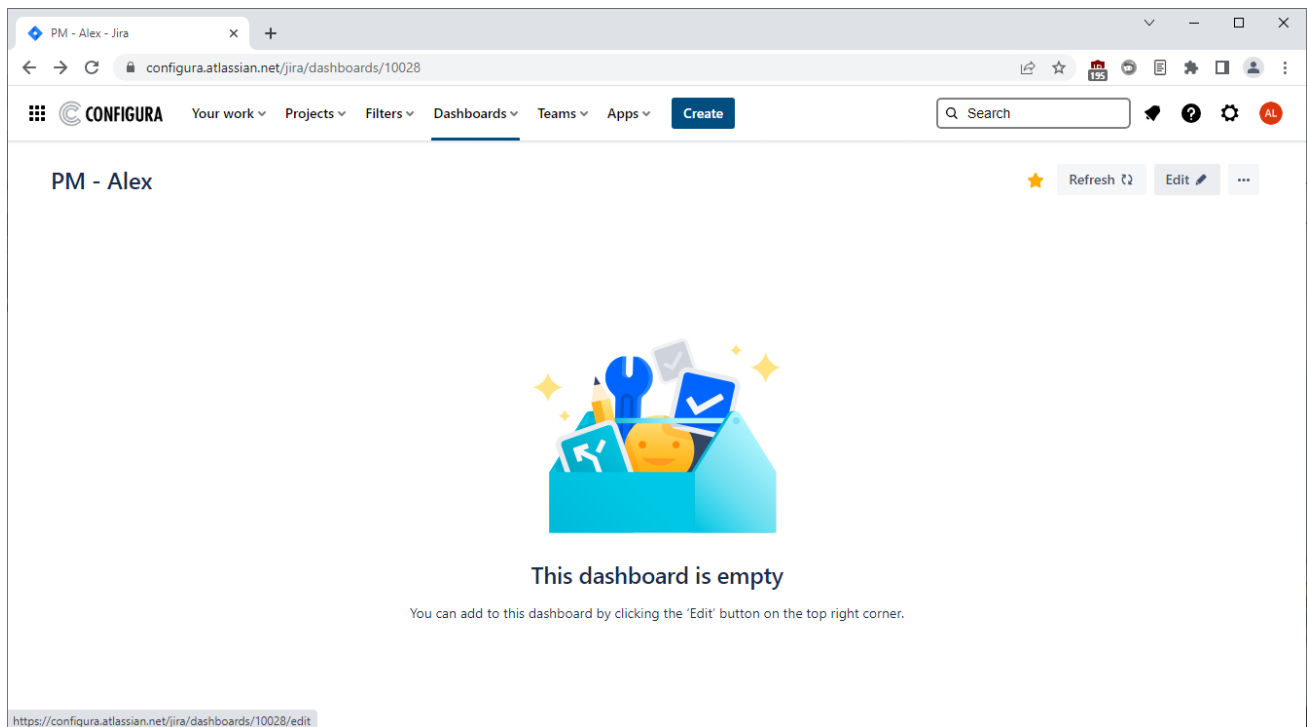
To add a gadget, click Add below the gadget you want to add:



Configure and save your gadgets. Once in place, a gadget can easily be moved by clicking the gadget header and dragging it to the desired location on the dashboard. Each gadget can also be edited:



To add or change gadgets later, press the Edit button:



## Backlog View

The **Backlog** view is where to:

- Create sprint versions.
- Change the sprint version for the issues.
- Start and close the sprint.
- Make an original time estimate for the sprint.

The backlog can be reordered by clicking and dragging the issues.

Clicking an issue in the backlog bring up the **Issue Detail View**, as shown here:

The screenshot displays the Jira interface for the 'Esther's Fika' project. The left sidebar shows navigation options under 'PLANNING' (Board, Timeline, Backlog, Active sprints, Reports) and 'DEVELOPMENT' (Issues, Components, Releases, Project pages, Risks). The main area is titled 'Backlog' and shows a list of issues grouped by 'EPIC Sprints'. The selected issue, 'this is a user story 4382482394' (FD-52035), is highlighted. The right sidebar shows the 'Issue Detail View' for this issue, including its title, status (Closed), description, and linked issues. The issue is a user story with a time estimate of 50h. The backlog list includes various tasks and user stories, some with time estimates and others with checkboxes. The interface is clean and modern, with a light blue and grey color scheme.

## Board (Kanban) / Active Sprint (Scrum) View

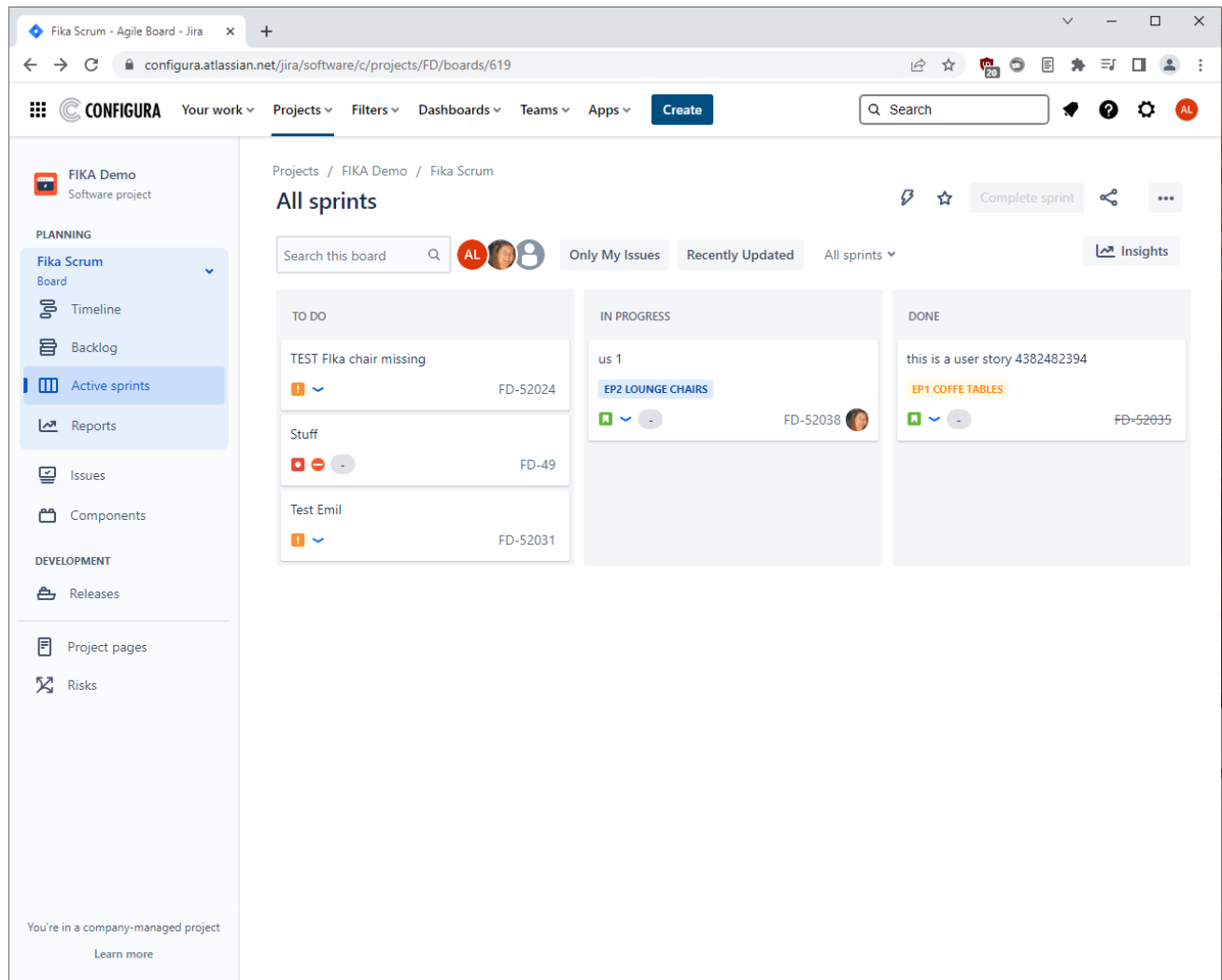
The **Board/Active Sprint** view is where the development team do their daily work. There can be multiple boards for different scrum teams or groups (currently set up by Configura), and a single board can cover multiple projects.

The Board/Active Sprint has a card layout where **Columns**, **Swimlanes** and **Quick Filters** group and filter the cards.

*Example: Board (Kanban)*

The screenshot shows the FIKA Kanban board in Jira. The interface includes a top navigation bar with 'CONFIGURA' and various menu items like 'Your work', 'Projects', 'Filters', 'Dashboards', 'Teams', and 'Apps'. A search bar is also present. The left sidebar shows the 'FIKA Demo' project and a list of items under 'PLANNING' (FIKA Kanban Board, Timeline, Kanban board, Reports) and 'DEVELOPMENT' (Issues, Components, Releases, Project pages, Risks). The main area displays the 'Kanban board' for 'FIKA Demo'. It features three columns: 'TO DO 30', 'IN PROGRESS 3', and 'DONE 0'. Issues are organized into swimlanes: 'Expedite 2 issues' (containing 'Stuff' and 'this is a ladybug'), and 'Everything Else 31 issues' (containing 'CET Designer', 'Test Emil', 'Stuff', 'CET Designer', and 'TEST Fika chair missing'). A right sidebar shows a list of issues with details like 'us 1', 'EP2 LOUNGE CHAIRS', 'analyse problem', and 'update model 220'. A message at the bottom right states 'We're only showing recently modified issues. Looking for an older issue?'.

### Example: Active Sprint (Scrum)



### Scrum vs Kanban

#### Scrum

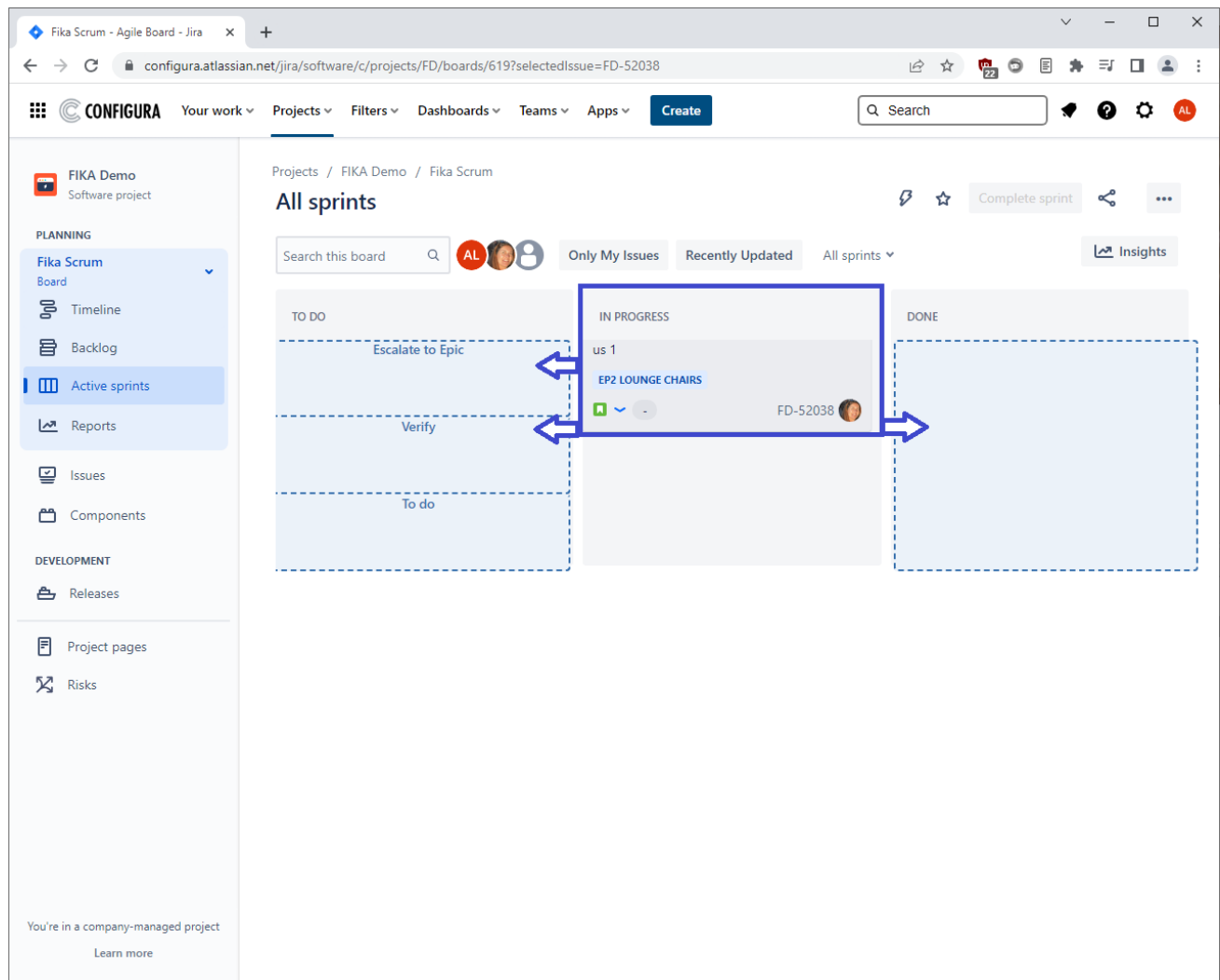
- Create sprint backlogs
- Add tasks to sprints
- More iterative

#### Kanban

- Work directly from master backlog
- Pull next highest priority from the master backlog
- Less iterative

## Changing Issue Status in the Board/Active Sprint View

When working in the **Board/Active Sprint** view, issue status is changed by dragging the card from one column to another. When doing so, Jira will guide you to where the card may be moved:



## Releases View

The **Releases** view lets you know the status of a version at a glance. Hover over a progress bar for a summary, or click a version to see the complete status, including the issues, development data, and potential problems.

For each version, it is possible to generate a basic release notes draft\* via the Release Notes links as highlighted in the screenshot below:

The screenshot shows the Configura web interface for a project named 'FIKA Demo'. The left sidebar contains navigation options under 'PLANNING' (FIKA Kanban Board, Timeline, Kanban board, Reports) and 'DEVELOPMENT' (Issues, Components, Releases, Project pages, Risks). The 'Releases' option is selected. The main content area is titled 'Phase 1' and includes a 'Release notes' link highlighted with a blue box. Below this is a 'Description' section with a placeholder 'Add a description...'. A 'Related work' section shows a prompt to 'Start adding some related work' with an 'Add related work' button. The 'Issues' section displays a list of issues with their IDs, descriptions, and status (CLOSED, NEW, WORK IN PROGRESS, VERIFIED). The right sidebar shows 'Unreleased' information, including start and release dates, and a 'Progress' section with a bar chart and a summary of issue counts: Done: 1 issue, In progress: 2 issues, To do: 4 issues, Warning: 0 issues.

| Issue ID | Description                     | Status           |
|----------|---------------------------------|------------------|
| FD-52035 | this is a user story 4382482394 | CLOSED           |
| FD-52036 | this is a user story 490        | NEW              |
| FD-52038 | us 1                            | WORK IN PROGRESS |
| FD-52041 | this is my test task            | VERIFIED         |
| FD-52042 | analyse problem                 | WORK IN PROGRESS |

\*Pre-requisites to make this work are:

1. A version has to be created in the Project Settings.
2. A Fix Version has to be specified for each issue to determine which release it should be included in.

When release notes are generated using this button, they will include text that has been entered into issues' Release Notes field:

The screenshot displays the Atlassian Configura web interface for a user story titled "this is a user story 490". The interface includes a left sidebar with navigation options like "FIKA Demo", "PLANNING", "Issues", and "DEVELOPMENT". The main content area shows the "Release" tab selected, with a "Release notes" text field containing the text "These are release notes". Below this, there are sections for "Linked issues", "Designs", and "Activity". The right sidebar contains a "Details" panel with various fields such as "Assignee", "Reporter", "Development", "Releases", "Labels", "Team (migrated)", "Found In", "Impact", "Fixed in", "Developer", "QA", "External Reference", "Original estimate", and "Epic Link".

Projects / FIKA Demo / FD-52034 / FD-52036

### this is a user story 490

Attach Create subtask Link issue ...

General Time Testing Version VisionFlow **Release**

**Release notes**

These are release notes

Linked issues

Clones

FD-52035 this is a user story 4382482394

**Designs**

https://figma.com/file/... Add Design

Paste the URL for a public Figma file to attach it to this JIRA issue. Learn more

**Activity**

Show: All Comments History Emails Time Balance Newest first

Add a comment...

Pro tip: press M to comment

**Details**

Assignee Unassigned Assign to me

Reporter Esther Puente

Development Create branch Create commit

Releases Add deployment Add feature flag

Labels None

Team (migrated) None

Found In None

Impact None

Fixed in None

Developer None

QA None

External Reference None

Original estimate 100h

Epic Link EP1 Coffe tables



In the release notes, the text input appears like this:

The screenshot shows a web browser window with the URL `configura.atlassian.net/projects/FD/versions/10614/tab/release-report-all-issues`. The application interface includes a top navigation bar with 'Your work', 'Projects', 'Filters', 'Dashboards', 'Teams', and 'Apps' menus, along with a 'Create' button and a search bar. A left sidebar lists project components like 'FIKA Demo', 'FIKA Kanban Board', 'Timeline', 'Kanban board', 'Reports', 'Issues', and 'Components'. The main content area displays the 'Create release notes' dialog. This dialog has a 'Create in' section with 'Confluence' and 'This version' options, a 'Preview' section showing a list of release notes, and a 'Formatting options' section. The preview shows a title 'Release notes - FIKA Demo - Phase 1' followed by a 'Story' section containing three items: 'FD-52035 this is a user story 4382482394', 'FD-52036 this is a user story 490' (highlighted with a blue box), and 'FD-52038 us 1'. Below the story is a 'Task' section with 'FD-52044 try this' and 'FD-52041 this is my test task'. At the bottom of the dialog are 'Copy to clipboard' and 'Save release notes' buttons.

FIKA Demo: Phase 1 - Jira

configura.atlassian.net/projects/FD/versions/10614/tab/release-report-all-issues

CONFIGNRA Your work Projects Filters Dashboards Teams Apps Create

FIKA Demo Software project

PLANNING

FIKA Kanban Board

Timeline

Kanban board

Reports

Issues

Components

DEVELOPMENT

Releases

Project pages

Risks

Create release notes

Create in

Confluence This version

Preview

Formatting options

Release notes - FIKA Demo - Phase 1

Story

FD-52035 this is a user story 4382482394

FD-52036 this is a user story 490

Release notes

These are release notes

FD-52038 us 1

FD-52044 try this

Task

FD-52041 this is my test task

Copy to clipboard Save release notes

## Reports View

The **Releases** view offer many different report templates for reports, charts, and diagrams.

The screenshot shows the Configura Jira Reports View interface. The browser address bar displays the URL: `configura.atlassian.net/projects/FD?selectedItem=com.atlassian.jira.jira-projects-plugin%3AReport-page`. The interface includes a top navigation bar with the Configura logo, a search bar, and a 'Create' button. A left sidebar lists navigation options: 'FIKA Demo Software project', 'Back to project', 'Reports', 'All reports' (selected), 'AGILE', 'FIKA Kanban Board', 'Cumulative Flow Diagram', 'Control Chart', 'ISSUE ANALYSIS', 'Average Age Report', 'Created vs Resolved Issues ...', 'Pie Chart Report', 'Recently Created Issues Re...', 'Resolution Time Report', 'Single Level Group By Report', 'Time Since Issues Report', 'FORECAST & MANAGEMENT', 'Time Tracking Report', and 'You're in a company-managed project'. The main content area is titled 'All reports' and displays a grid of report templates under the 'Agile' section. The templates include: 'Cumulative Flow Diagram' (a stacked area chart), 'Control Chart' (a line chart with data points), 'Issue analysis' (a bar chart), 'Average Age Report' (a bar chart with a trend line), 'Created vs. Resolved Issues Report' (a line chart with two series), and 'Pie Chart Report' (a pie chart). Each template has a brief description of its purpose.

A few examples of available reports are:

- **Sprint Report:** understand the work completed or pushed back to the backlog in each sprint.
- **Burndown Chart:** track the total work remaining and project the likelihood of achieving the sprint goal.
- **Velocity Chart:** track the amount of work completed from sprint to sprint.
- **Control Chart:** shows the cycle time for your product, version, or sprint.
- **Cumulative Flow Diagram:** shows the statuses of issues over time. This helps you identify potential bottlenecks that need to be investigated.

## Issues View

The **Issues** view lists all open issues in the board but can be filtered to show only certain issues. This can be done by premade filters or by using a query for advanced searches:

The screenshot displays the Jira Issues view for the 'FIKA Demo' software project. The interface includes a top navigation bar with tabs for 'Your work', 'Projects', 'Filters', 'Dashboards', 'Teams', and 'Apps'. A search bar is located in the top right corner. On the left side, there is a 'Filters' sidebar with options like 'All issues', 'My open issues', 'Reported by me', 'Open issues', 'Done issues', 'Viewed recently', 'Resolved recently', 'Updated recently', and 'View all filters'. The main area shows a list of issues under the 'Issues' tab. A search bar at the top of the list contains the query 'project = "FD" ORDER BY created DESC'. The list of issues includes 'old discontinued fika files in the setting up CET designer course' (FD-52058), 'top parts' (FD-52057), 'Leg parts' (FD-52056), 'Table parts' (FD-52055), 'Table' (FD-52054), 'Epic issue' (FD-52053), and 'picture does not show'. The detailed view of issue FD-52058 is shown on the right, with the title 'old discontinued fika files in the setting up CET designer course'. The issue is in the 'Release' state and has no release notes. The 'Linked issues' section shows a link to issue FD-52058 with the status 'CLOSED'. The 'Details' section shows the assignee as 'Unassigned'. At the bottom, there is a comment box with the text 'Add a comment...' and a 'Pro tip: press M to comment' message.

The list can be ordered in different ways:

The screenshot shows the Jira interface for a project named 'FIKA Demo'. The main view is 'Issues', and the current issue is 'FD-52058: old discontinued fika files in the setting up CET designer course'. The issue is in the 'Release' tab, with 'Release notes' and 'Linked issues' sections. A dropdown menu is open, showing 'ORDER ISSUES BY' options: Created (selected), Key, Last viewed, Priority, Resolved, Status, and Updated. The dropdown is highlighted with a blue box. The issue details show it is assigned to 'Unassigned' and has a status of 'CLOSED'.

Projects / FIKA Demo

Issues

Share Export issues Go to advanced search LIST VIEW DETAIL VIEW

project = "FD" ORDER BY created DESC Save filter BASIC JQL

1 of 38

FD-52058

old discontinued fika files in the setting up CET designer course

Attach Create subtask Link issue

New Actions

General Time Testing Version VisionFlow Release

Release notes

None

Linked issues

relates to

ES-201976 FD-52058: old discontinued fika files in the settin... NR CLOSED

Details

Assignee Unassigned

Add a comment...

Pro tip: press M to comment

Selecting an issue in the list shows detailed information about it in the **Issue Detail** view:

The screenshot displays the Atlassian Jira web interface. The top navigation bar includes the 'CONFIGURA' logo, a search bar, and various project management tools like 'Your work', 'Projects', 'Filters', 'Dashboards', 'Teams', and 'Apps'. A 'Create' button is also present.

On the left sidebar, the 'FIKA Demo' project is selected. Below it, there's a 'Back to project' link and a 'Filters' section with options like 'All issues', 'My open issues', 'Reported by me', 'Open issues', 'Done issues', 'Viewed recently', 'Resolved recently', and 'Updated recently'. A 'View all filters' link is at the bottom of the sidebar.

The main content area shows the 'Issues' list for the 'FIKA Demo' project. The list is filtered by 'project = "FD" ORDER BY created DESC'. The first issue, 'old discontinued fika files in the setting up CET designer course' (FD-52058), is highlighted. Below it are other issues like 'top parts' (FD-52057), 'Leg parts' (FD-52056), 'Table parts' (FD-52055), 'Table' (FD-52054), and 'Epic issue' (FD-52053). A 'picture does not show' message is visible at the bottom of the list.

The 'Issue Detail' view for FD-52058 is shown on the right. It includes the issue title, a description, and a 'Release notes' section. The 'Linked issues' section shows a link to issue 'ES-201976' with the status 'CLOSED'. The 'Details' section shows the assignee as 'Unassigned'. At the bottom, there's a comment box with the text 'Add a comment...' and a 'Pro tip: press M to comment' message.

The issue status can be updated in the **Issue Detail** view using the **Status** button. Simply click the button to open the dropdown and choose a status to update to that status:

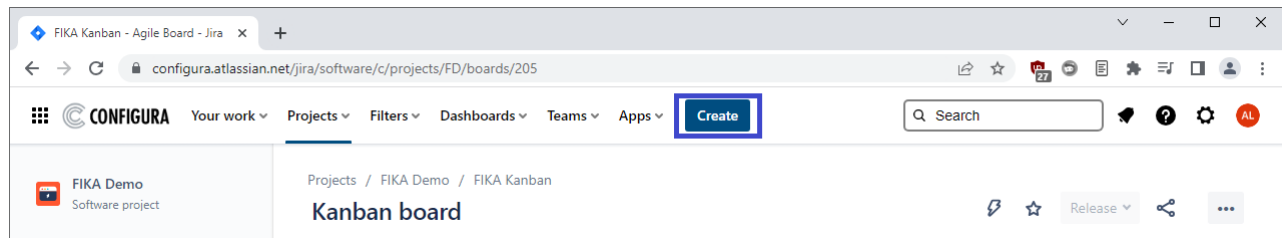
The screenshot shows the Jira issue detail view for 'this is a user story 490' (FD-52036). The 'New' dropdown menu is open, showing status options: Verify (VERIFIED), More info needed (MORE INFO NEEDED (VERIFICATION)), Close (CLOSED), Verification in progress (VERIFICATION IN PROGRESS), and View workflow. The left sidebar shows filters, and the top navigation bar includes 'CONFIGURA' and 'Your work'.

If you need to communicate directly with a user who does not have a Jira account, you can use the **Email** button available in the **Actions**-menu on **Issue Detail** view. If the user replies to the email, the reply is added to the issue as a comment.

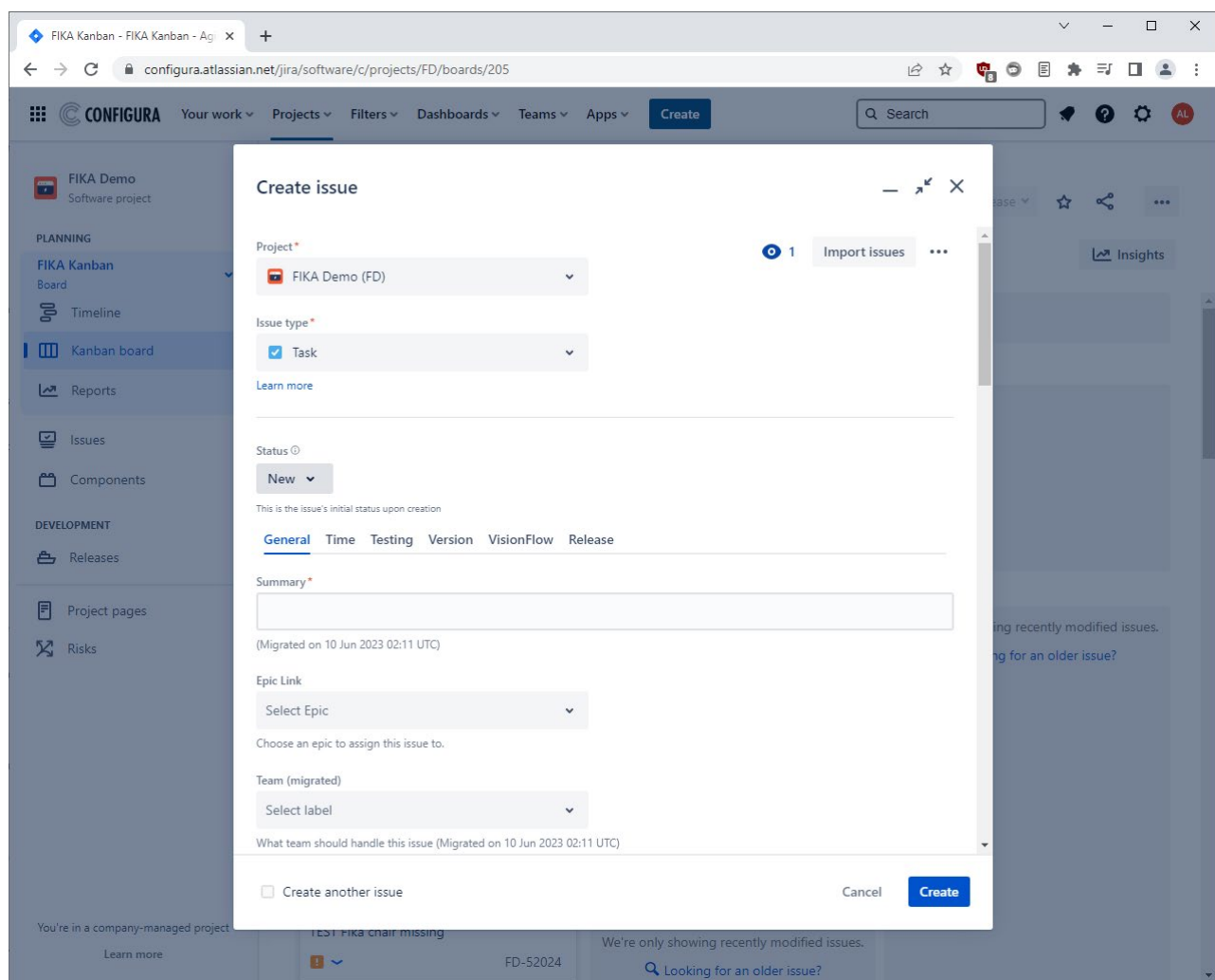
The screenshot shows the Jira issue detail view for 'this is a user story 490' (FD-52036). The 'Actions' dropdown menu is open, showing options: Add flag, Email, Log work, Connect Slack channel, and Create subtask. The left sidebar shows filters, and the top navigation bar includes 'CONFIGURA' and 'Your work'.

## Creating a new Issue

To create a new issue, click the **Create Issue** button on the menu bar, or use the keyboard shortcut C:



The template consists of many fields, and the more information that you include the better. The fields **Project**, **Issue Type** and **Summary** (a.k.a Title) are compulsory and cannot be left empty, and we also recommend to always add a **Description**:



Components and Labels are optional but very useful for sorting and filtering issues. Components are categories available in the board, while Labels are free text that can be applied to an issue.

## Issue Types

The following **Issue Types** are available:

### Create Issue

The screenshot shows the 'Create Issue' form with the following fields and options:

- Project\***: A dropdown menu with 'Racking (RAC)' selected.
- Issue Type\***: A dropdown menu with 'Task' selected. A mouse cursor is hovering over the dropdown, which has opened to show a list of options: Epic, Story, New Feature, Bug, Regression Bug, QA, Crash, User Bug Report, and User Feature Request. The 'Epic' option is highlighted at the top of the list.
- Summary\***: A text input field.
- Epic Link**: A text input field.
- Team**: A text input field.

Below the 'Team' field, there is a note: 'What team should handle this issue'.

Epics can have Stories and Tasks linked to them while Stories and Tasks can have Sub-tasks.

Please note that **User Bug Reports** and **User Feature Requests** always come in through support, and it is the responsibility of the Product Owner to review them, decide if action should be taken and, in that case, convert them to regular issues.



## Moving an Issue

You can move issues to another project or issue type. To do so, click the **Actions** menu and select **Move**

The screenshot displays the Jira Configura web application interface. The browser address bar shows the URL `configura.atlassian.net/browse/FD-49`. The top navigation bar includes the Configura logo, a search bar, and various navigation links like 'Your work', 'Projects', 'Filters', 'Dashboards', 'Teams', 'Apps', and a 'Create' button. The left sidebar contains a project overview for 'FIKA Demo' with sections for 'PLANNING' (Kanban Board, Timeline, Kanban board, Reports) and 'DEVELOPMENT' (Issues, Components, Releases). The main content area is titled 'Stuff' and shows the 'Release' tab selected. It includes sections for 'Release notes', 'Designs' (with a Figma link), and 'Activity' (showing comments). On the right, an 'Actions' menu is open, listing various options. The 'Move' option is highlighted with a red rectangle. Other visible options in the menu include 'Add flag', 'Email', 'Log work', 'Connect Slack channel', 'Convert to Subtask', 'Clone', 'Delete', 'Actions menu', 'Print', 'Export XML', 'Export Word', 'Take a tour', and 'Find out more'.

In the next step, move the User Bug Report to Bug (or User Feature Request to New Feature):

**Move Issue**

- Select destination project and issue type
- Map statuses
- Update fields
- Confirm changes

Select destination project and issue type

Your issues' destination project may have different settings than your source project. Even if they're named the same, issue types may use different workflows and fields. You may lose data when moving issues between projects.

[Learn more about migrating issues](#)

Select Project

Current Project: FIKA Demo → New Project: FIKA Demo (FD)

Select Issue Type

Current Issue Type: Bug → New Issue Type: Bug

Next Cancel

## Editing an Issue

Issues can be edited directly in the issue or in the popup view from the **Backlog**, **Board / Active Sprint**, and **Issues** views.

**FD-49**

Stuff

Attach Create subtask Link issue

General Time Testing Version VisionFlow Release

Description

Normal text B I ... A ...

Can I do this

Save Cancel

Designs

<https://figma.com/file/...> Add Design

Paste the URL for a public Figma file to attach it to this JIRA issue. [Learn more](#)

Activity

Show: All Comments History Emails Time Balance Newest first

Add a comment...

Pro tip: press **M** to comment

Details

Assignee: Unassigned [Assign to me](#)

Reporter: Jeremy Bakalar

Development: [Create branch](#) [Create commit](#)

Releases: [Add deployment](#) [Add feature flag](#)

Labels: None

Team (migrated): None

Found In: None

Impact: None

Fixed In: None

Developer: None

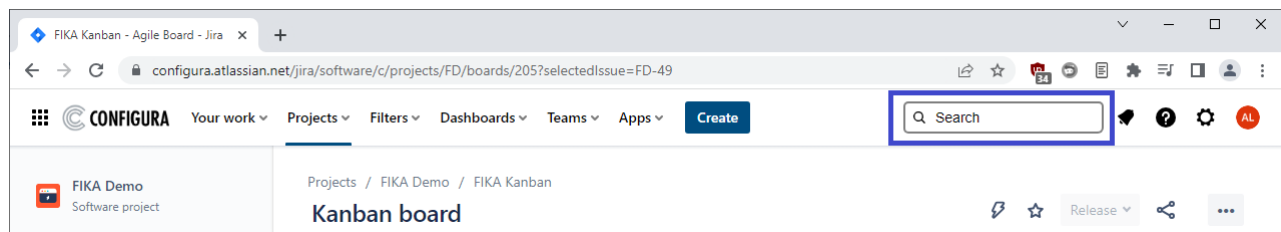
QA: None

External Reference: None

Note that the test result of an issue is found on the **Testing** tab.

## Searching for Issues

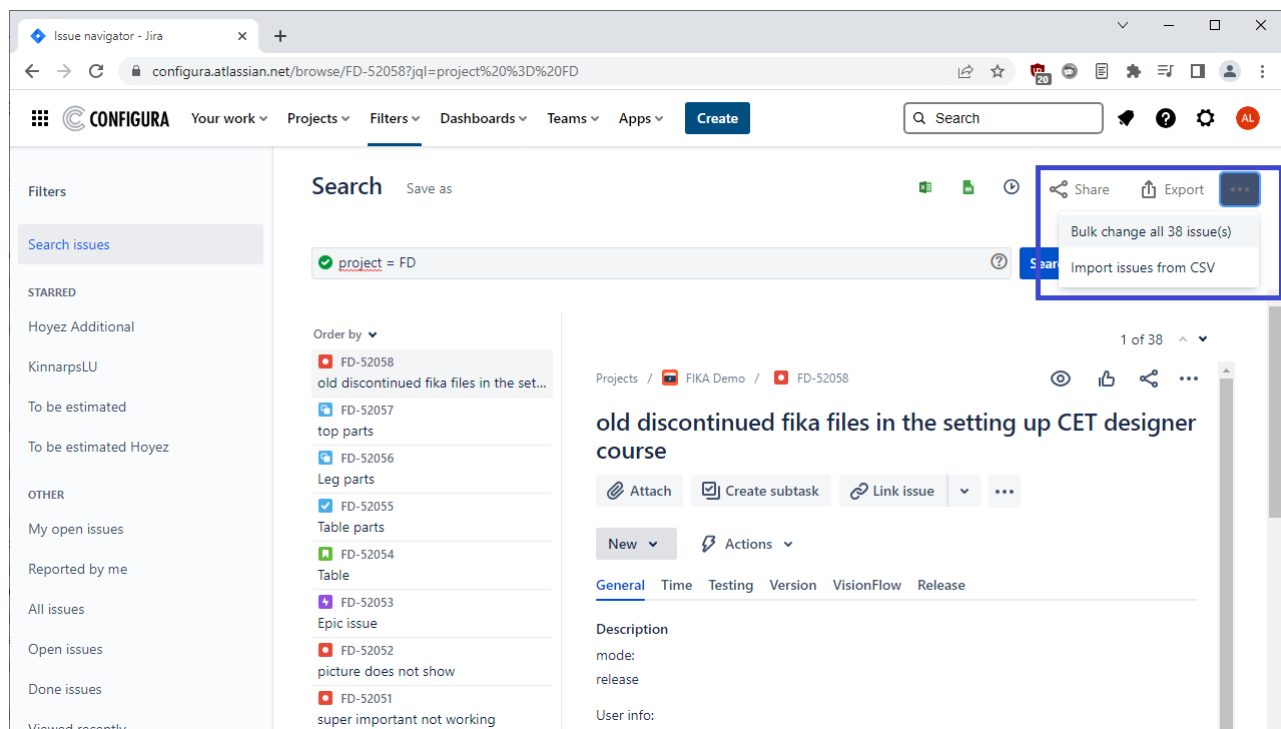
A **Quick Search** is always available, regardless of where you are in Jira:



The search is dynamic, and the results will update as you continue typing.

Please note that when searching, the list may look incomplete. Keep writing and more suggestions will be displayed.

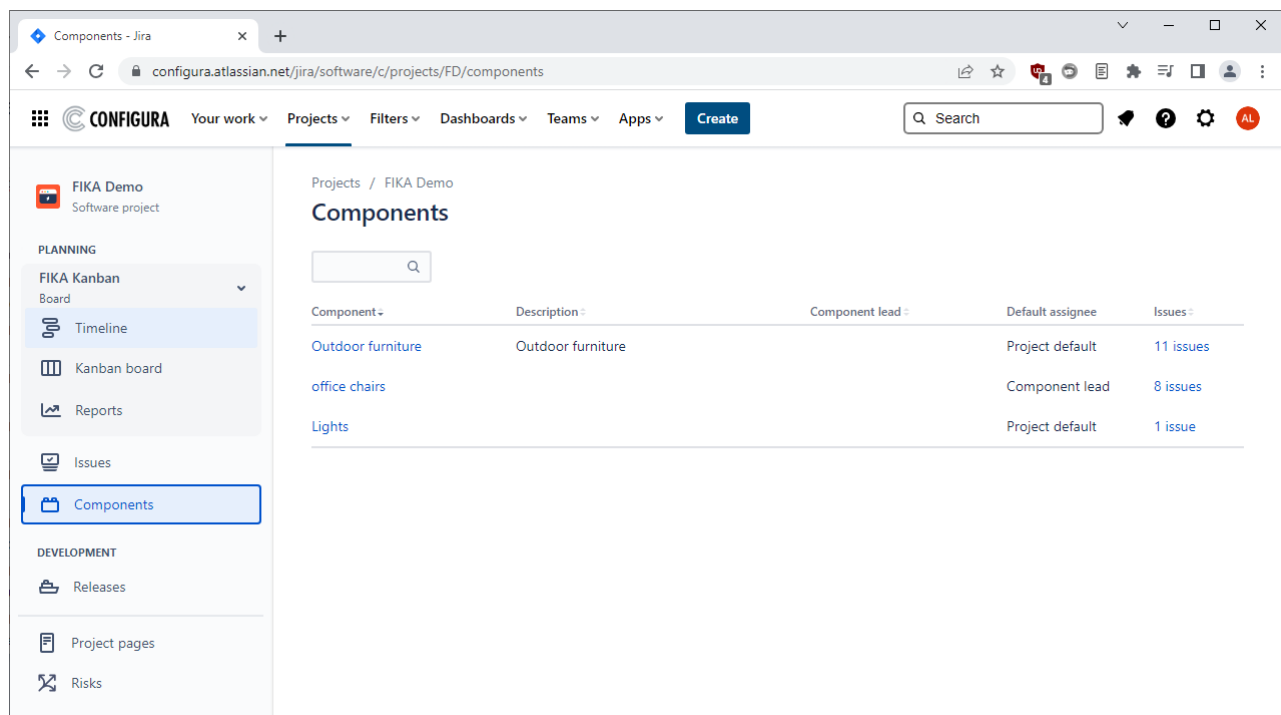
From the **Search** view, it is also possible to make bulk changes to multiple issues at once:



Bulk change is also possible by selecting multiple issues in the Backlog (CTRL click), right-clicking and selecting Bulk Change.

## Components View

The **Components** view is where you create and edit components, i.e. categories in the Board:



The screenshot shows the Jira web interface for the 'FIKA Demo' software project. The left sidebar contains navigation options under 'PLANNING' (FIKA Kanban Board, Timeline, Kanban board, Reports) and 'DEVELOPMENT' (Issues, Components, Releases, Project pages, Risks). The 'Components' option is highlighted. The main content area is titled 'Components' and displays a table of project components.

| Component         | Description       | Component lead | Default assignee | Issues    |
|-------------------|-------------------|----------------|------------------|-----------|
| Outdoor furniture | Outdoor furniture |                | Project default  | 11 issues |
| office chairs     |                   |                | Component lead   | 8 issues  |
| Lights            |                   |                | Project default  | 1 issue   |

Using Components is optional but very useful for sorting and filtering purposes.

## Submitting issues from CET Designer directly to partner's project

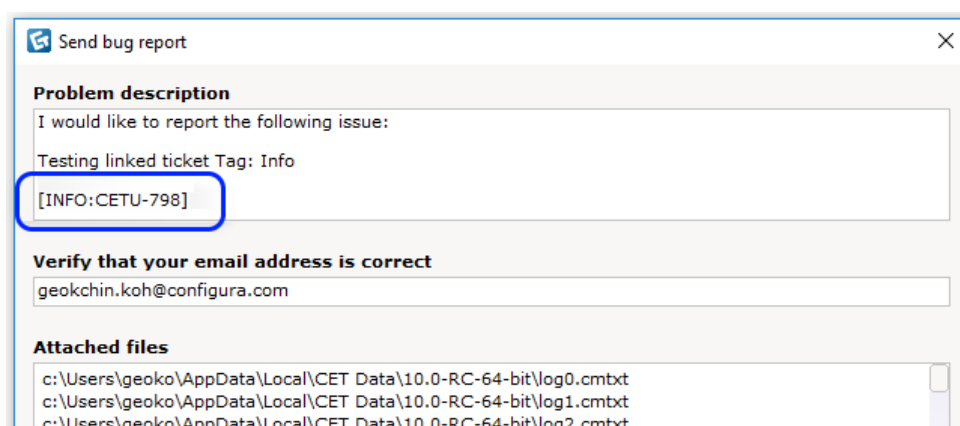
By default, all bug reports will be sent to the CET Support service desk, but there are cases where the partner's development team want to see their issues directly and not wait for the CET Support to escalate the issue.

If a development team (developers or QA:s) or UAT testers want to submit a bug report via CET Designer, they can choose to add the text **[INFO: <linked issue key>]** in the description field, as shown in the screenshot below. The linked issue key could, for example, be the parent test ticket number.

If you include this tag however, **[INFO: <issue key>]**, it will:

- Create a new bug ticket and link to the mentioned "linked issue key". Both tickets are in the same project.
- Remove the issue from the CET Support service desk.

The link type can be updated, removed, etc. as needed.



**Send bug report**

**Problem description**

I would like to report the following issue:

Testing linked ticket Tag: Info

**[INFO:CETU-798]**

**Verify that your email address is correct**

geokchin.koh@configura.com

**Attached files**

c:\Users\geoko\AppData\Local\CET Data\10.0-RC-64-bit\log0.cmtxt  
c:\Users\geoko\AppData\Local\CET Data\10.0-RC-64-bit\log1.cmtxt  
c:\Users\geoko\AppData\Local\CET Data\10.0-RC-64-bit\log2.cmtxt

In Jira, it will look like this:

### Linked issues



more info by



FD-52058

old discontinued fika files in the setting up CET designer c...



NEW

## Appendix 1: Global Project Automation Rules

### Jira Software Project

| SET THE STATUS TO               | AUTOMATION / WHAT WILL HAPPEN  | REMARK                        |
|---------------------------------|--|-------------------------------|
| New                             | Not available.   | Assignee will not be changed. |
| More info needed (verification) | Show a pop-up dialog for users to set an Assignee.   |                               |
| More info added (verification)  | Set Assignee to Developer or remove Assignee if developer field is empty.  |                               |
| Verified                        | Not available.   |                               |
| To Do                           | Not available.   |                               |
| Work in progress                | If the developer field is set, set Assignee to Developer. If not, set both Assignee and Developer to current user.   |                               |
| More info needed (WIP)          | Show a pop-up dialog for users to set an Assignee.   |                               |
| More info added (WIP)           | Set Assignee to Developer or remove Assignee if the developer field is empty.  |                               |
| Code review                     | Remove Assignee.   |                               |
| Ready for test (Dev)            | Remove Assignee.   |                               |
| Test in progress (Dev)          | Set Assignee to current user.  |                               |
| More info needed (Dev)          | Set Assignee to Developer.   |                               |
| Test passed (Dev)               | Not available.   |                               |
| Waiting for merge               | When it is transitioned from <i>Test in Progress (Dev)</i> , set QA field to current user.   |                               |
| Waiting for build               | When it is transitioned from <i>Test in Progress (Dev)</i> , set QA field to current user.   |                               |
| Ready for test (build)          | Remove Assignee.   |                               |
| Test in progress (build)        | Set Assignee to current user.  |                               |
| More info needed (build)        | Set Assignee to Developer.   |                               |
| Test rejected                   | Set Assignee to Developer.<br>When it is transitioned from <i>Test approved</i> , remove Resolution.<br>When it is transitioned from <i>Test In progress (build)</i> , set QA field to current user. |                               |
| Test approved                   | Remove Assignee.   |                               |

|                   |   |  |
|-------------------|---|--|
|                   | <p>When it is transitioned from <i>Test in progress (dev)</i> or <i>Test in progress (build)</i>, set QA field to current user.</p> <p>Set Resolution to Done.</p> <p>Set Remaining Time to zero.</p>   |  |
| Ready for release | Set Resolution to Done.   |  |
| Released          | <p>When the fix version is set to <i>Released</i>, change the issues to Released if the previous status is <i>Ready for Release</i>.</p> <p>Set all duplicate issues with <i>Closed</i> status to Released too.</p> <p>Send out an email or shared comment with customer to the reporter in the Service Desk ticket saying that an update has been released.</p> <p>Close the Service Desk ticket with resolution Done.</p> |  |
| Closed            | <p>Set Remaining Time to zero for all resolutions except for Revisit.</p> <p>Add Resolution info as a comment in the linked Jira Service Desk issue.</p>  |  |

| ACTION                          |   |   |
|---------------------------------|---|---|
| Add a comment with @support tag | If the previous status of the Jira Software ticket is <b>New</b> or <b>More info added (verification)</b> then it will be set to <b>More info needed (verification)</b> . | No changes for other statuses (Verified, Closed, etc.). |
| Edit priority                   | Add a comment "Escalated issue got priority.." on the Service Desk issue.   |   |

## Jira Service Desk

| SET THE STATUS TO | AUTOMATION   | REMARK   |
|-------------------|--|--|
| Close             | If the previous status of the Jira Software ticket is <b>More info needed (verification)</b> , then it will be set to <b>More info added (verification)</b> automatically. | No changes for other statuses (New, Verified, Closed, etc.). |

| ACTION                            |   |  |
|-----------------------------------|---|--|
| Add a comment with @developer tag | If the previous status of the Jira Software ticket is <b>More info needed (verification)</b> or <b>Closed</b> , then it will be set to <b>More info added (verification)</b> automatically. | No changes for other statuses (New, Verified, Closed, etc.).<br><br>Comment will still be copied to the Jira Software project. |
| Update resolution                 | Add a "Support issue got resolution:" comment in the Jira Software ticket.  |  |



## Appendix 2: Jira Software Keyboard Shortcuts

| Keyboard Shortcuts                 |                                     |
|------------------------------------|-------------------------------------|
| <strong>Global Shortcuts</strong>  |                                     |
| Go to Dashboard:                   | <b>g</b> then <b>d</b>              |
| Next Item:                         | <b>j</b>                            |
| Browse to a Project:               | <b>g</b> then <b>p</b>              |
| Go to Boards:                      | <b>g</b> then <b>a</b>              |
| Find Issues:                       | <b>g</b> then <b>i</b>              |
| Previous Item:                     | <b>k</b>                            |
| Administration Quick Search:       | <b>g</b> then <b>g</b>              |
| Go to Tempo:                       | <b>g</b> then <b>t</b>              |
| Log work with Tempo:               | <b>w</b>                            |
| Quick Search:                      | <b>/</b>                            |
| Create an Issue:                   | <b>c</b>                            |
| Quick Operations:                  | <b>.</b>                            |
| Open shortcut help:                | <b>?</b>                            |
| Form Submit:                       | <b>Ctrl</b> + <b>Alt</b> + <b>s</b> |
| <strong>Navigating Issues</strong> |                                     |
| View selected Issue:               | <b>o</b> or <b>Enter</b>            |
| Next Issue:                        | <b>j</b>                            |
| Previous Issue:                    | <b>k</b>                            |
| Dock/Undock the Filters Panel:     | <b>[</b>                            |
| Next Activity:                     | <b>n</b>                            |
| Previous Activity:                 | <b>p</b>                            |
| Focus search field:                | <b>f</b>                            |
| Search for issues:                 | <b>u</b>                            |
| Switch filter view:                | <b>t</b>                            |
| Detail View Order by:              | <b>z</b>                            |
| Log work on issue:                 | <b>w</b>                            |
| <strong>Issue Actions</strong>     |                                     |
| Edit Issue:                        | <b>e</b>                            |
| Assign Issue:                      | <b>a</b>                            |
| Comment on Issue:                  | <b>m</b>                            |
| Share Issue:                       | <b>s</b>                            |
| Edit Issue Labels:                 | <b>l</b>                            |
| <strong>Board Shortcuts</strong>   |                                     |
| Backlog:                           | <b>1</b>                            |
| Active sprints / Kanban board:     | <b>2</b>                            |
| Reports:                           | <b>3</b>                            |
| Presentation mode:                 | <b>z</b>                            |

(Source: <https://www.atlassian.com/blog/jira-software/4-ways-get-jira-keyboard-shortcuts>)