

Clear*vision*

NINJA GUIDE TO



⚡ Jira Service Management

03	INTRODUCTION
04	GETTING STARTED
06	CONFIGURING YOUR PROJECT
08	GETTING SLAS SORTED
09	QUEUES AND REPORTING
10	AUTOMATING SIMPLER TASKS
12	KNOWLEDGE BASE
14	WORKING WITH TICKETS
16	ADVANCED LEARNING



CONTENTS

INTRODUCTION

Welcome to our Ninja Guide to Jira Service Management (JSM), written by ITIL certified subject matter experts with 20+ years of experience in ITSM.

You're here to get the most out of your investment in Jira Service Management and to give your customers experiences they'll remember for all the right reasons, but first...

What is JSM?

Built on the Jira platform, when used correctly, JSM is a powerful tool primarily used by IT teams that is becoming increasingly popular amongst business functions, e.g. HR, Finance, and Legal, where requests are frequent and often dealt with over email.

If you work in any of these areas, you'll be pleased to know that with JSM, you can receive, track, manage, and resolve requests from your team's customers — customers can send requests by email, a customisable help centre, and an embeddable widget.

Jira Service Management makes it easier to categorise service requests, incidents, problems, and changes by organising and prioritising requests in a single place, keeping your team on track with goals or Service Level Agreements (SLAs).

Whilst this guide is designed to help you get started with the tool that saw Atlassian named a '[Leader in The Forrester Wave™: Enterprise Service Management, Q4](#)' and 'the only Visionary in the 2021 Gartner Magic Quadrant for ITSM Tools', it does not replace the comprehensive [training courses](#) we offer.

Our experts are trusted by thousands of Atlassian customers as a result of the successful services delivered, read about one such [case here](#). Now, it's your turn. We have tons of free content on our website for you to take advantage of, such as our [tour of Jira Service Management webinar](#), where Atlassian Solution Architect Gary Blower covers the latest additions to the product and more in less than 20-minutes.

Why not follow us on social media for the latest content and updates from the Atlassian world?



GETTING STARTED

To make the most of JSM projects, there are a few things to consider...

First, decide on the use and 'customers'. Customers can be anything from external users assisted by an IT Support team to internal users within a Legal or HR department. Next, decide how you want your customers to interact with the team to determine who has access to the project and the available methods. Then, review the types of requests and the information required to do the job. The best course of action is to look at previous requests and build a catalogue, which is effectively an open list of work customers can browse and choose from. A well-defined request catalogue helps:

- Reduce the load by identifying requests that take the most resources, either by automating laborious tasks or improving documentation.
- Collect all of the necessary information on a request form to complete a job. If it's a new user/starter, a description box won't help, instead, consider creating specific fields to capture important info such as the start date, line manager, team, etc.
- Automate task opportunities in the future.

Kick this off with a spreadsheet — create a table like the one shown below to collect all of the information. For details like possible applications or departments, provide a list:

REQUEST NAME	INFORMATION NEEDED/LIKED TO COMPLETE THE WORK	MANDATORY FIELD
Getting error when accessing a system	Summary	Yes
	Description	Yes
	What System? <ul style="list-style-type: none">• Jira• Confluence• Bitbucket• StatusPage• Opsgenie	Yes
	Error code	No
	Attachment	No
Get access to a system	What system?	Yes
	Justification	Yes
	Line Manager	Yes

Now you can create your first JSM project! Start by going into Jira, click 'projects' at the top, and 'create project'. You will then be prompted for the 'type' of project. *Note: You may need an Admin to do this for you.*








Templates are a great starting point; create basic workflows and insert common requests for the project type. *Tip: Click on each arrow for a breakdown of the template features.*

Project templates

Jira Service Management

Deliver great service experiences fast. Empower every team, from IT to HR to legal, to set up a service desk quickly and continuously adapt at scale with our service

-  **IT service management** LAST CREATED
Handle service requests, resolve incidents, approve changes and fix problems using ITSM best practices. >
-  **General service management**
Manage all your service requests in one location and help your employees get the answers they need. >
-  **Customer service management**
Deliver great service experiences fast with a template designed to help your external customers. >
-  **HR service management**
Manage onboarding and offboarding, answer questions, and facilitate change requests for your staff. >
-  **Facilities service management**
Easily manage requests for maintenance, moving, and event planning. >



CONFIGURING YOUR PROJECT

Once you've created the project, you can configure the settings.

The following changes are made in the 'Project Admin'. Speak to a System Admin for assistance.

CUSTOMER PERMISSIONS

To configure customer permissions, go into the project and select 'project settings' in the bottom left-hand corner. Click 'customer permissions' towards the top to assign who can raise tickets for the project and who can share them. If used, 'Organizations' provides a collection of customers visibility over tickets.

CHANNELS

Channels are options that connect customers. Choose from 'email', 'customer portal', 'widget', or a combination.

EMAIL

To provide customers with an email to raise requests, head to the 'email request' tab halfway down 'project settings'.

On Cloud, Atlassian provides a default email, but if you'd rather use your own, choose the appropriate button at the bottom of the page and follow the on-screen instructions.

Once connected, you can link to a request type. To do this, click '...' towards the top on the right-hand side and select 'configure email request types'. Then, select the request type you wish to use and remember to save it.

WIDGETS

If you'd like quick access on your website service desk or another location, you can use widgets. To configure this, select the 'widget' tab under 'project settings'. Using toggles, colour picker, and text boxes, configure the widget how you like and save and copy the JavaScript code to the appropriate space. For the widget to work, your product must allow non-logged-in users to raise requests.

CUSTOMER PORTAL

Your portal is set up automatically when you make a JSM project. It is the primary way for customers to interact and allows information to be collected. Under the 'portal settings' tab, configure colours and define permissions, i.e. who can add announcement banners.



PORTAL REQUESTS

Once you've set up your portal, configure the different requests previously defined. Create custom fields for additional data to be captured. If you're not an Admin, speak to the relevant person about getting custom fields made and added to the screens for you.

Once all custom fields and request types have been created, group them so that the portal remains clean and user-friendly. *Note: You are limited to just two levels — 'group' and 'request' forms, so think wisely about what makes sense to your customers.*



GETTING SLAS SORTED

A service-level agreement (SLA) is a documented contract between a service provider and its customers outlining services, standards of practice, and deadlines. In short, SLAs are commitments. *Reminder: Customers can be external or internal users.*

In Jira terms, services are request forms, standards are Agent practices, i.e. how they will interact, the processes, etc. Deadlines are the SLA clocks set in Jira, for example, triage and resolution times. To create SLA clocks, head to 'project settings' and click 'SLAs'. *Note: Some SLAs will be in the project by default, but you can edit or create new ones if preferred.*

Each SLA has four parts — 'goals', 'start parameter', 'finish parameter', and optional 'pause parameters'.

Goals

Issues will be checked against this list, top to bottom, and assigned a time goal based on the first matching JQL statement.

Time Goal	Calendar	Issues to display (in JQL)	
⋮ 4h	📅 Sample 9-5 Calendar	"Ticket category" = Incidents	✕
⋮ 8h	📅 Sample 9-5 Calendar	"Ticket category" = "Service requests"	✕
No target	📅 Sample 9-5 Calendar	All remaining issues	

+

Conditions

Time will be measured between the start and stop conditions below.

Start counting time when...

Issue Created

Resolution: Cleared

Add Condition

Pause counting time during...

+

Finish counting time when...

Resolution: Set

Add Condition

When creating a goal, set the time and clock you wish it to run on, and use Jira Query Language (JQL) to show the issues that will use that goal. Next, use the start, finish, and pause parameters to decide when the clock runs. To create additional calendars, click the 'calendar' icon towards the top on the right-hand side. Give it a name and default time zone to run against, and populate each day with start and end times. You can also insert holidays, if known, to show the availability of your team. *Note: Compulsary holidays do not need editing, however, moved dates must be adjusted.*



QUEUES AND REPORTING

Queues are a way of providing a focus on tickets as they reveal what's waiting to be triaged, who is waiting on support, etc. Queues, like SLAs, are built using JQL.

To build queues, go into the project, click 'queues' at the top, and 'manage queues'. When the pop-up appears, select 'create a new queue', give it a name, and start building!

You can use buttons to help build your query, or you can select 'switch to JQL' to build it from scratch. Lastly, choose the columns you wish to show on the queue. You have access to every custom field available in the project and are not limited by the number of columns you can have, but bear in mind, the purpose is to provide Agents with an understanding of the ticket before they go into it.

Alongside queues, Agents and Service Managers can use JQL to create in-depth reports. To do so, click 'reports' when in the project.

Jira will generate tons of different reports by default, such as created vs resolved, time to resolution, SLA success rate, etc. You can also create your own. To do this, click 'new report' in the bottom left, give the report a name, and then 'add series'.

Each series is made up of a data point, i.e where Jira collects issues for the report, its name, colour, and the JQL behind it. *Note: You can add as many series as required to the report.*

You can continue using normal Jira filters on dashboards alongside project reports. For additional dashboard gadgets that display information in a specific way, [Custom Charts for Jira by Old Street Solutions](#) is highly recommended and comes with a selection of styles.



AUTOMATING SIMPLER TASKS

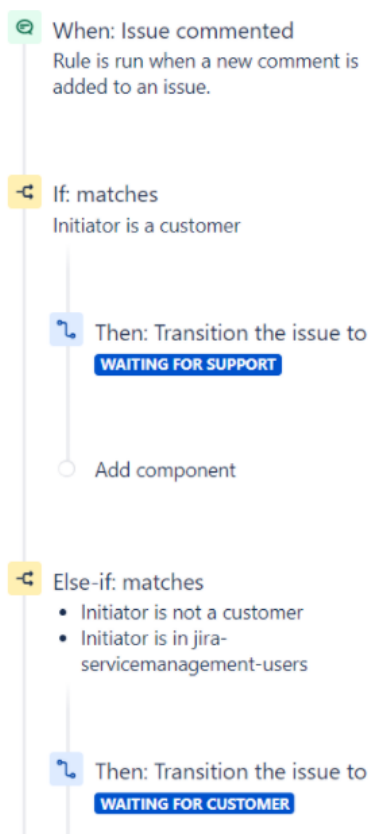
Tasks such as moving tickets to wait for the customer, looking for replies, alerting resolver groups, moving tickets to support, or adding issue data based on form details, can all be automated. *Automation for Jira*, formerly owned by *Code Barrel*, offers Project Admins no-code automation rules that have all the power without the technical debt.

Each rule requires a trigger and at least one action. Typical trigger points are when issues are created, commented on or transitioned, and where fields are edited or run on a schedule. Actions are creating, editing, commenting or transitioning issues. You can send notifications to external chat tools such as *Teams* or *Slack* and even external webhooks as actions.

TRANSITION ON COMMENTS

Whenever anyone comments on a ticket, the status should reflect its current state by default.

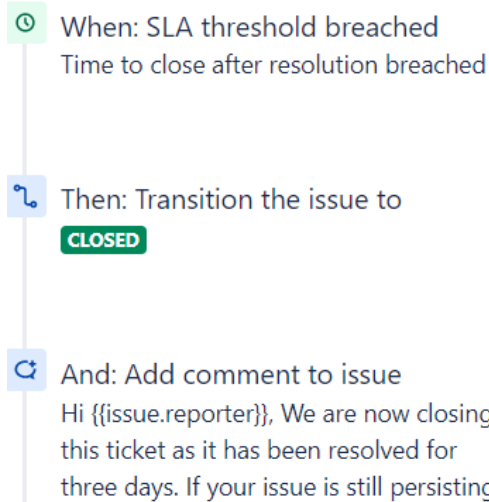
To begin, you need the trigger point to be 'issue commented'. Next, apply some logic. As you are looking for two opportunities, apply a condition, in particular, the 'if else' condition. Identify whether the user is an external customer using the 'user condition' option, and if they are, the action is to transition the issue to 'wait for support'. If they are not a customer, use the user condition option again to see if they are a part of the Jira-servicemanagement-user group. Finally, transition the issue to 'wait for the customer'. If done correctly, the rule should look something like this...



AUTO CLOSE

If you're giving customers the ability to reopen tickets after they have been resolved, but you would like them permanently closed after so many days, set up 'pseudo-SLAs'. Using the SLA feature, set clocks to run, and use them as trigger points for the automation rules to kick in. For this, create an SLA that starts when a ticket is resolved and ends when it enters the closed status.

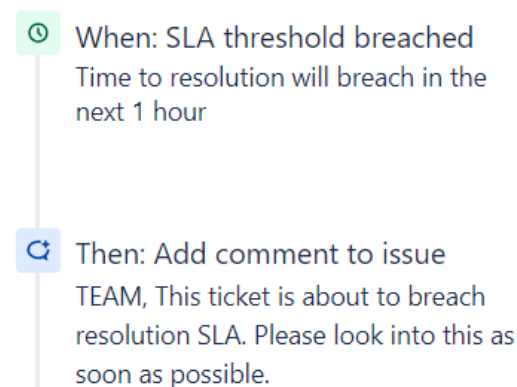
Remember to set a goal on tickets. Using the automation rule, set the trigger to 'SLA breached'. Specify the SLA, and select when the timer is breached, then transition the ticket to closed. Use this opportunity to notify the customer by commenting on the issue.



SLA BREACHING

To stay on top of SLAs, set rules to notify you when a ticket is close to breaching or has breached one of the clocks so that you can deal with it promptly. For rules such as this, the trigger point is an 'SLA breach'. Select the rule and time you wish to be notified and choose an action, for example, 'send a chat notification to *Teams* or *Slack*, and this will add a comment to the ticket or direct @ mention to users.

The hardest part of any automation is knowing where to start. *Jira Automation* comes with a diverse and deep template library. If you are struggling to start a rule, look in the library and adapt an existing one to meet your needs.



KNOWLEDGE BASE

A knowledge base article contains information about a product or service either in the form of a user guide or to solve common problems. Typically, a service desk will fill a knowledge base with anything that can assist the customer to deflect tickets. This is often product release notes, Admin and user guides, how-to articles, and troubleshooting articles.

Knowledge bases empower customers by allowing them to fix their own issues. If articles are well written, customers often prefer to fix requests themselves over raising a ticket and waiting for a response. Knowledge base articles can be shown on the portal to customers without the need for a license. When a customer types a request, JSM automatically filters the correct knowledge base articles that could be used to help deflect the ticket.

Summary *

Suggested articles


 **test project**
from test project

Description

If you have both Confluence and JSM as Cloud editions, the project will automatically create a linked space for you. To change the space used or to add additional spaces, go to 'project settings' and select 'knowledge base'. Once connected, you can decide whether you want it to be public or specifically for Agents.

Space

Search for linked spaces

<input type="checkbox"/> Space	Viewing permissions		
<input type="checkbox"/> test project	Private <input type="button" value="v"/>	<input type="button" value="Unlink"/>	<input type="button" value="Settings"/> 

Control article suggestions

Set up article suggestions that can help your customers as they fill in a request form. Use labels to limit which articles are shown in each form.



To filter articles, select specific labels against request types and ticket deflection.

Control article suggestions

Set up article suggestions that can help your customers as they fill in a request form in each form.

Request form	Allow suggestions	Only show articles
Get IT help	<input checked="" type="checkbox"/>	atlassian x
Fix an account problem	<input checked="" type="checkbox"/>	

To see how successful an article is, every JSM project comes with two built-in. Head to 'reports', 'requests deflected', and 'requests resolved'. You can also see how often and when an article was last used in a project under the 'knowledge base' tab.

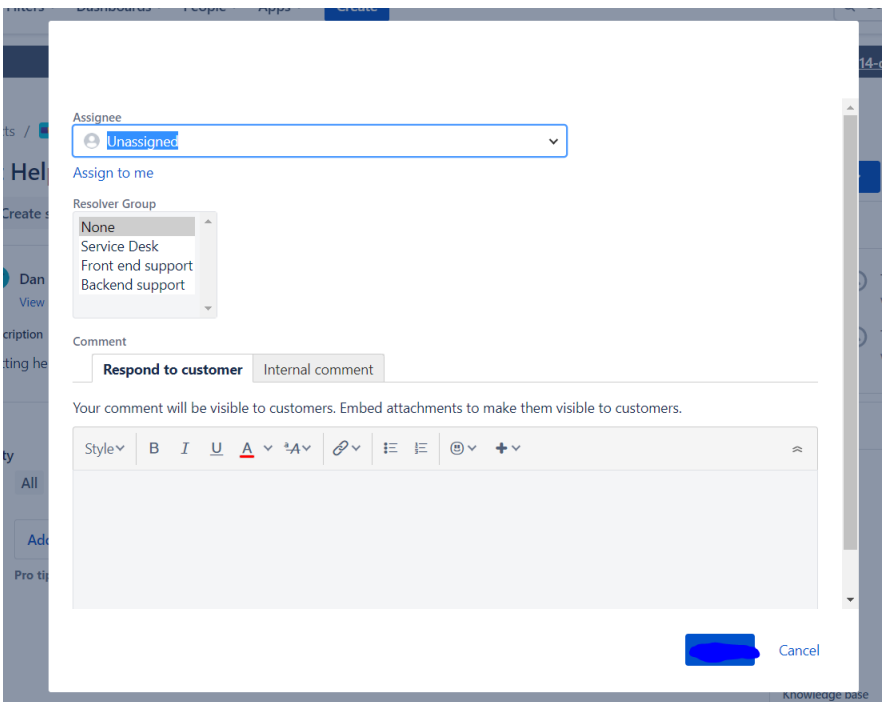
KEY POINTS

- o Labelling pages is not only best practice in Confluence, but it helps get articles onto a service portal.
- o To hide internal pages, use 'page restrictions'. If you're only allowing certain requests to use specific labels, a page can still be found via the 'help center'.
- o You can have multiple knowledge bases linked to a single portal. This allows knowledge management by separate teams.



WORKING WITH TICKETS

If your service desk has multiple resolver groups, an easy way to manage tickets for teams is to create a multi-select custom field. This will allow you to make queues for each team so that they can find their tickets at a glance. To do this, select 'resolver group' in the drop-down menu. If you know certain request types on the portal will be assigned to particular resolver groups, don't forget to add the field to the request type and to hide it with the pre-set value. Finally, to move tickets between teams, either change the value in the resolver group field or create a looped transition.



It's common for tickets to be escalated when additional support is required from Managers. While every company's escalation policy differs, there is a simple way of having Jira show escalated tickets — update workflows to include an escalated status. By using a status reflected on the customer portal, customers are kept informed on the progress of their requests. Typically, this status can only be transitioned to 'from waiting for support' and will usually return to 'waiting for support' or 'resolved'. This simple workflow change enables escalations to be time stamped on a ticket for future analysis.

For teams to know what tickets have been escalated, create a queue that looks for "status = escalated".

Filter by

This queue is already filtered by: project = "test project"

Type: All ▾ Escalated ▾ Unresolved ▾ Label: All ▾ + More

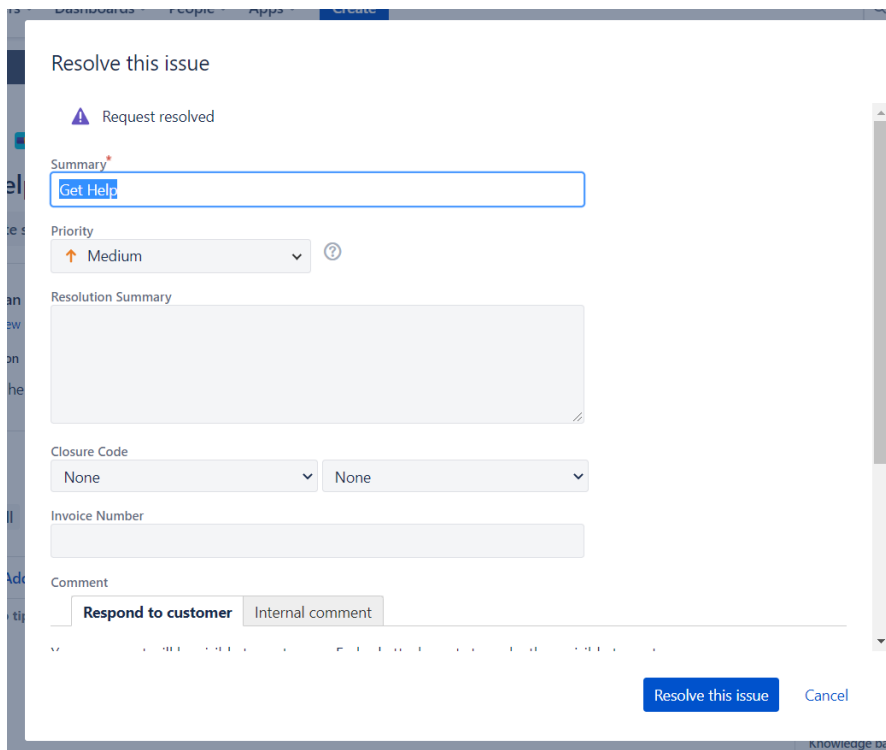
Columns

Issue Type × Key × Summary × Reporter × Assignee × Status × Created ×



Resolving tickets is a great feeling, but learning from them benefits everyone. Adding extra fields when resolving a ticket, enables teams to review and understand previous requests within seconds. 'Resolution summaries' are a great way to provide future Agents with an understanding of what a ticket is and how best to resolve it. It's also a great opportunity to look into 'closure codes'. Closure codes provide Managers with an overview of how a ticket was resolved so that if there was a technical problem, future occurrences can be mitigated. To set this up, create two new custom fields; the first should be resolution summary (a multi-line text field is best). The second should be for closure codes (single select or cascade select - depending on how in-depth you need closure codes to be).

Once done, create a new 'resolve ticket screen' and add the fields. Lastly, add your screen to the resolve transition on the workflow.



The screenshot shows a 'Resolve this issue' form. At the top, there is a status indicator 'Request resolved'. Below that is a 'Summary' field containing the text 'Get Help'. The 'Priority' is set to 'Medium'. There is a 'Resolution Summary' multi-line text area. Below that is a 'Closure Code' dropdown menu set to 'None'. There is an 'Invoice Number' field. At the bottom, there is a 'Comment' section with two tabs: 'Respond to customer' (selected) and 'Internal comment'. At the bottom right, there are two buttons: 'Resolve this issue' and 'Cancel'.

Triaging tickets to completion allows Agents to understand them in their entirety for the best course of action. For all the details, check the summary and description of the request/incident and determine whether the reporter has logged the ticket correctly. Has the correct request type been chosen? Has it been logged as an incident? Has it got the correct priority based on the impact and urgency? Has any of the custom field data been populated/need to be populated on the ticket? Have the correct values been set?

While triaging correctly is time-consuming, it reduces ticket resolution times as Agents know what to look for. It also helps Managers understand where best to place resources — if more incidents are logged, Managers can allocate resources to quickly investigate and mitigate any challenges before more tickets are created.



ADVANCED LEARNING

Between advanced searching and creating and managing projects, there's [a lot to learn about Jira](#). As outlined at the start of this white paper, you'll want to consider an advanced course to expand your understanding. Courses are available at the Admin level for new and experienced users.

WHAT'S INCLUDED IN CLEARVISION'S ADVANCED COURSES?

Jira is a powerful ticketing system with many features including 'asset management' and 'event management'. We urge you to take advantage of one of our advanced courses as listed below.

JIRA SERVICE MANAGEMENT ESSENTIALS (AVAILABLE FOR [SERVER, DATA CENTER, AND CLOUD](#))

This course covers Atlassian Jira concepts and will equip you with everything you need to raise service requests, manage service desk queues, SLAs, and customer satisfaction levels.

JIRA SOFTWARE/SERVICE MANAGEMENT ADMINISTRATION (AVAILABLE FOR [SERVER, DATA CENTER, AND CLOUD](#))

This course is ideal for new Jira Administrators who have the responsibility of defining projects and configurations in agile teams.

[JIRA PROJECT ADMINISTRATION \(SERVER AND DATA CENTER\)](#)

Based on the Atlassian certification ACP-600, attendees will gain a solid understanding of both the configurations in their control and the ramifications of requesting changes from an Application Administrator. Attendees will learn best practices of shared schemes, permissions, and project configurations.

All of our training is customisable, meaning you can pick and choose modules from other courses, and we'll tailor and develop them in a way that suits you.

[JIRA MENTORING \(SERVER, DATA CENTER, AND CLOUD\)](#)

This is mentoring on a one-on-one basis with a certified Atlassian Solution Architect.

Okay you will have to add the line at the end because I don't want to mess up your design. Please paste this at the very end of the paper: For a visual overview of Jira Service Management, [watch our tour](#).

