Creating and maintaining a backlog of different types of Jira issues needs a considerable amount of time and effort. Sometimes the work you have to do related to the tasks and workflows becomes repetitive. Some examples include adding sub-tasks for a support engineer's query, assigning an issue to a user, changing the due dates, automating the development tasks using a source code management tool, etc. Jira gives you the option to use rules to do these manual and repetitive tasks automatically. This Jira automation tutorial' describes how you can create Jira automation rules and what scenarios you can use them in.

What are Automation Rules in Jira?

JIRA software has a rule builder for automating tasks by configuring automation rules. This allows you to automate common and repetitive processes and workflows and enables you to focus on the most important work.

Triggers, Conditions, and Actions

Jira automation rules consist of triggers, conditions, and actions. The trigger starts off the automation rule by listening to events like issue creation or changing a particular parameter. Jira gives you the option to specify the trigger the way you want and configure it to run on a schedule.

Jira conditional statements such as 'if block', 'else block', and 'JQL condition' allow you to create more specific rules for certain conditions. An example of a conditional scenario is sending an email only if the issue is a priority 1 (P1) issue and the environment is production.

Actions specify what you need to do. Actions enable you to automate tasks and processes. Examples of actions include creating sub-tasks for an issue, sending an email, etc.

How to Create Automation Rules in Jira

1.Go to your project in Jira and click on the "Project settings" link, which appears at the bottom of the left-hand side navigation bar. Then select 'Automation".

2. Click the "Create rule" button in the top right-hand corner of the automation page.

Credit: Atlassian

3. Next, select "Trigger". Jira automation lists out triggers available based on different components such as Issue triggers, DevOps, Integrations, Scheduled, Quick Actions, Jira service management, etc. Select the issue type you want and hit save.
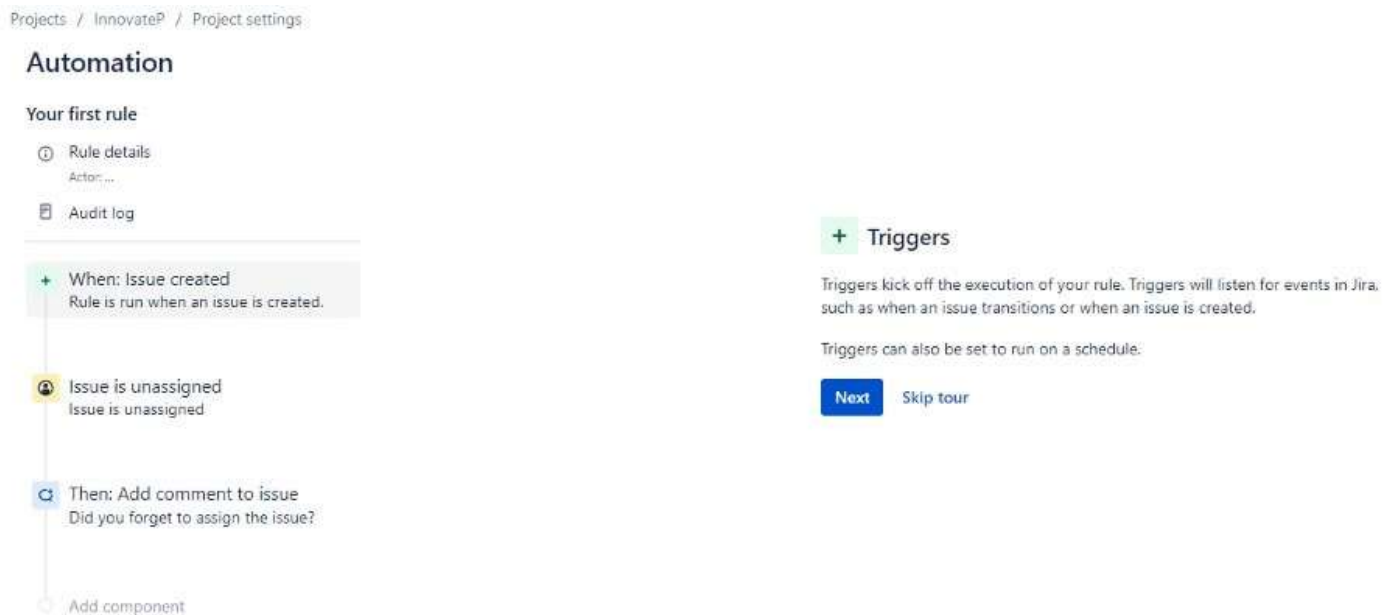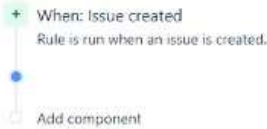

Credit: Atlassian

4. Next, select the appropriate component from the new branch, new action, and new condition. These components allow you to control the automation's execution.

Credit: Atlassian

5. Once you have configured the component, fill out the appropriate information and hit the save button.

6. Name your rule and hit 'Turn it on.' If you want, you can edit the rule. For example, you can allow only admins to edit rules.

Credit: Atlassian

7. After creating the rule, it will appear in the rules tab of the automation page. If you want to edit the rule, click on the rule and then edit all the steps. After making the changes, hit "Publish changes."

Add component

New branch
Create a separate section of this rule and perform actions and conditions on other items.

New action
Actions perform changes to a system.

New condition
Actions will only execute if all conditions preceding them pass.

Publish rule

OR Publish changes

Credit: Atlassian

Once created, you can see the full audit log, which shows triggers, actions, and conditions you used it in. The rule section shows the rule's execution status.

Automation

Create-Sub-Tasks
- Rule details
- Audit log

+ When: Issue created
Rule is run when an issue is created.

Issue Type equals
Story

Then: Create 4 sub-tasks

Audit log
1-4

Date	Rule	Status	Duration	Operation
14/07/22 08:56:34 pm (9100952211)	Create-Sub-Tasks	CONFIG CHANGE		Show more
14/07/22 08:55:02 pm (9100913163)	Create-Sub-Tasks	CONFIG CHANGE		Show more
14/07/22 08:43:34 pm (9100613779)	Create-Sub-Tasks	CONFIG CHANGE		Show more
14/07/22 08:42:30 pm (9100585431)	Create-Sub-Tasks	CONFIG CHANGE		Show more

> What do the different statuses mean?

Credit: Atlassian

How to Create Automation Rules for Common Tasks

Automatically Creating Sub-tasks for an Issue

Suppose you are working on a software development project epic. For most user stories of agile software development teams, you need to create common sub-tasks tested by the developer. In that case, automate sub-tasks creation rather than creating them every time you make a user story.

1. Create a new rule
2. Next, select the trigger 'when the issue is created.'
3. Then, you have the option to add a condition. For example, automate the sub-task creation only if the issue type is a user story.
4. Next, select the create sub tasks action.
5. Finally, specify all the sub-tasks you want to create and hit save.

Automation

Create-Sub-Tasks

- ⓘ Rule details
- ▤ Audit log

+ **When: Issue created**
   Rule is run when an issue is created.

⤨ **Issue Type equals**
   Story

▣ **Then: Create 4 sub-tasks**

○ Add component

Jira lets you further expand this rule to more complex rules such as adding a sub task match status, or adding an assignee after creating the subtasks, etc. Look at the audit log of the automation rule you created, and add any component if you want to edit the rule.

Auto-send an Email When Issue Priority Increased

Suppose your project needs to escalate high-priority bugs and features as soon as you log them into Jira. In that case, the easiest thing is to create an automation rule to notify the appropriate users about their priority and make them take the necessary action on time without any delay. Here is an example of how you can auto-send emails for a change in issue.

1. Create a new rule with `create rule`.
2. Next, select your trigger. Here you monitor a specific issue field, the 'Priority' field. Therefore, select the 'Field value changed' trigger and the priority field. Specify the change type. For example, if you want to trigger the rule whenever there is a change in this field. In our case, it is a change in the priority field. Next, select for which operations you need to trigger it.
3. Next, add a condition. Select the 'JQL (Jira Query Language) condition' and write your JQL to match your condition. In our case, we send an email only if the priority becomes high. Therefore, the query should be 'Priority > High'.
4. Finally, select your action as 'Send email'. Configure the action, including the email subject, content, and who received the email. You can also specify the sender details as well as whether to send it in plain text or HTML format.

When specifying the configurations, make use of automation smart values. Smart values enable accessing issue data. For instance, say you want to include the issue summary in your email body. In that case, you can utilize the smart value '{{issue.summary}}', which returns the issue summary you enter when creating the issue. Therefore, make use of smart values to create many intuitive automation rules.

The following is the audit log of the automation rule you just created. If you want to update the automation rule add any component to it.

P1 Issue created

� **When: Value changes for**
Priority

🔍 **If: Issue matches JQL**
priority > High

✉ **Then: Send email**
Assignee, Reporter
P1 Issue created

○ Add component

Auto-assign the User When you Create an Issue

Auto assigning issues is another feature in Jira that is easy to do using a Jira automation rule.

1. Create a new rule.

2. Select your trigger as 'When the issue is created'.

3. Straight away, you can add the 'Assign issue' action.

4. Specify and select the user you want to assign it to.

Projects / InnovateP / Project settings

Automation   NEW

ⓘ Rule details

+ When: Issue created
Rule is run when an issue is created.

Then: Assign the issue to
? Issue reporter

Add component

Jira not only lets you specify simple automation rules like the ones described above, but it also lets you specify more complex automation rules combining more conditions and actions. Also, Jira has a collection of automation templates that allow you to define more common automation tasks with minimal effort. Thus Jira is a great tool to create automation rules and save your valuable time to focus on more important work.