ATLASSIAN University

Jira Automation

Lab Workbook

Table of Contents

Jira Automation Introduction 1
Lab format 1
Lab 1 - Course Overview 2
Lab 2 - Automation Overview 4
Exercise 1 - Create a kanban project. 4
Exercise 2 - Create an automation rule that creates subtasks when creating an issue. 4
Lab 3 - Create More Rules 7
Exercise 1 - Create an automation rule that creates subtasks if the rule is manually executed. 7
Exercise 2 - Create an automation rule that auto-assigns high priority issues 8
Exercise 3 - Modify the rule to use an If / else block condition. 9
Lab 4 - Administration 14
Exercise 1 - Create a kanban project for testing rules. 14
Exercise 2 - Create a rule that writes to the audit log. 14
Exercise 3 - Limit groups that can trigger a rule 15
Exercise 4 - Copy a rule to another project 15
Exercise 5 - Explore global administration. 16
Lab 5 - Smart Values 18
Exercise 1 - Create a rule that lists the keys for all open issues in a project 18
Lab 6 - Advanced Rules 20
Exercise 1 - Create an automation rule that copies a comment from a child issue to the parent epic 20
Exercise 2 - (Optional) Create and use a rule with an incoming webhook trigger 21
Lab 7 - Jira Service Management Automation 23
Exercise 1 - View Jira Service Management automation 23
Exercise 2 - Create a new Jira Service Management automation rule 24

Jira Automation Introduction

Lab format

Optional exercises and appendices

The labs may have optional exercises. These are not required to complete the course. However, if you have time and interest, they supplement the exercises for the lab. There may also be appendices that you don't need to complete the class. They are full of useful information like additional reading and best practices. Dig into these after you complete the course!

Language and User Interface

The language you see in the Atlassian product UI is set to your browser's language. If you wish to see the UI in English (to match the lab instructions), or in a different language, go to your Atlassian user profile and edit your account preferences.

Cloud products are constantly being updated with new features, so you may see some slight differences between the lab instructions and the product you are using.

Logging in to your lab environment

To log in during the labs, you need your assigned site URL and the user's email address specified in the lab directions and password. If you're taking an On Demand course, you'll find these in the **Virtual Lab Instructions** activity in the **Lessons** section. If you're taking an instructor-led course, your instructor will share these details with you.

ⓘ The password for every user is the same. Keep this password easily accessible.

⚠ When switching between products in these labs, you can see other sites. It's important that you choose the product on the site that's been assigned to you.

Here's a list of the user(s) and what role they have in this course.

Name	Role
admin	Administrator
Alana Grant	Team member
Ryan Lee	Team member

Lab 1 - Course Overview

Estimated time: 5 minutes

Logging into your lab

To log in during the labs, you need an assigned **site URL** and each user's **email address** and **password**. If you're taking this as an OnDemand course, you'll find this in the **"Virtual Lab Instructions"** activity. If you're taking an instructor-led course, you'll receive the details from your instructor.

The **password** for **every user** is the **same**.

Do not log in with your own Atlassian ID

You probably already have an Atlassian account that you use to log in to your own Atlassian products. In the labs for this course, a specific set of users has been added to the cloud site. You will log in with these accounts. Do not log in using your own Atlassian ID.

Log in using a new browser, or an Incognito or Private window

A single browser can only handle one Atlassian account. This is because browsers keep cookies. Once you're logged into a cloud site on one browser, it remembers that login. So, if you open a new tab, you can't login as someone else.

To log into your lab, use a different browser to the one you usually use, or use an incognito or private window to log in.

Logging into labs as different Users

In the labs you'll need to log in as one or more users. To avoid logging in and out a lot you can either use different browsers or use an incognito or private window for each user.

Opening an Incognito window

> ⓘ You can open either an incognito window (Chrome) or private window (Firefox) from the browser menu. Other browsers also have the same functionality.

Chrome

1. Either:
    a. Click the Chrome three-dot (ellipses) menu button or
    b. From the Chrome browser menu click File.
2. In the dropdown menu, click New Incognito Window

Firefox

1. Either:
    a. Click the three-line Firefox application menu or
    b. From the Firefox browser menu click File.
2. In the dropdown menu, click New Private Window

Accessing your site

1. Use your assigned **site URL to navigate to your site.**

You're all set!
When you get to Jira/Confluence, you'll be told who to log in as.

Lab 2 - Automation Overview

Estimated time: 10 minutes

Exercises:

1. Create a kanban project.
2. Create an automation rule that creates subtasks when creating an issue.

Exercise 1 - Create a kanban project.

1. Navigate to the URL of your Cloud site and log in to Jira as **admin**.
    - Note: The **admin** user is a Jira administrator. Jira administrators can create company-managed projects.
2. If you see onboarding screens such as "Invite your team" and "Help us set up your Jira" screens, click **Skip**.
3. If:
    - you see "Project templates", "Create project" or "Add project details", you are already starting the process of creating a project. You can skip the next step and add the details as stated below.
    - you **do not** see any of those screens, you will need to create a project. Click the **Create project** button in the upper right (or in the top navigation, click **Projects > Create project**).
4. Click **More templates** to access all templates. In the “Project templates” screen, select **Software development**. Click the **Kanban** template. Click **Next** or **Use template** if needed.
5. Click **Show more** to display additional details. For project type, select **Company-managed project**. If you are unable to create a company-managed project, verify that you are logged in as **admin**.
6. In the **Add project details** window, enter **projectA** for the project name. You can leave the default **Key** as is. Verify that the template is **Kanban**.
7. Click **Next** or **Create project**.
8. On the Connect your project screen, select **Skip**.
   You should see the Kanban board for your project.
    a. If you see onboarding screens here such as "Invite your team" and "Select some tools..." screens, click **Skip or Continue**.

Congratulations, you have created a kanban project.

Exercise 2 - Create an automation rule that creates subtasks when creating an issue.

1. If the sidebar on the left is not expanded, click **>** (expand sidebar) to expand it. It is recommended that you keep the sidebar expanded for this lab.

2. You can get to automation two ways - through Project settings or using the Automation icon above the Kanban board on the right:
    a. To use the shortcut, click the Automation icon (a lightning bolt) above the Kanban board on the right and select **Manage automations**.
    b. To use Project settings, in the sidebar, click **Project settings** and then **Automation**.
        i. Note: You see the **Project settings** link because you created this project and therefore you are a Jira project administrator. By default, most of the project automation rules that you will create can be created by any Jira project administrator for the project. You could add another project administrator in Project settings by clicking **People** in the sidebar, then **Add people,** and then adding a user to the **Administrators** role.
3. You should see the **Automation** screen. This is where you can create automation rules for the project.
4. You should be brought to the **Rules** or **Templates** tab. If you land on the **Rules** tab, click on the **Templates** tab. This contains sample rules that you can explore and turn on for your project. Click on any library rules that interests you, then click **Return to templates** in the upper left. Click **OK** in the Discard changes dialog to confirm.
5. Return to the **Rules** tab on the **Automation** screen.
   Note: If you click the **Global administration** link, you will land on the Automation page within Global Administration. You can see this link because you're a Jira Administrator. If you want to create an automation that applies to all projects, you add rules here. In this case, we do not want to create a Global rule, so make sure you create the automation in ProjectA. Do not create the automation in the Global Administration area.
6. If you clicked the **Global administration** link:
    a. Navigate to Project A
    b. Click **Project settings**
    c. Click **Automation**
7. Click **Create rule** from ProjectA.
8. You can view the short "first rule" tour or click **Skip tour**. If you have completed the tour, click **Get started**.
9. Explore the triggers that are available to you. Triggers kick off a rule. They can be based on an event that occurs (most of the triggers), can be scheduled to run (see the "Scheduled" trigger), or manually triggered from an issue's details (see the "Manual trigger").
10. Create a trigger that executes when an issue is created:
    a. Under **Recommended**, click **Issue created**.
    b. Click **Save** or **Next**.
11. Create a condition that only selects created issues if they are of type **Task**:
    a. Under **Add component**, click **IF: Add a condition**.
    b. Select **Issue fields condition**.
    c. Under **Field**, select **Issue Type**.
    d. Under **Condition**, select **equals**.
    e. Under **Value**, select **Task**.
    f. Click **Save** or **Next**.

12. Create an action that automatically creates four subtasks when creating an issue:
    a. Under **Add component**, select **THEN: Add an action**.
    b. Select **Create sub-tasks**.
    c. Under **Summary**, type in **Write blog post**, then click **Add another sub-task**. (If you accidentally click Save to save the action, click on the action to edit it.)
    d. Repeat this process to create three more subtasks with summaries of **Create graphics**, **Approve blog post** and **Submit blog post**.
    e. Click **Save** or **Next**.
13. Verify that under **Automation**, your automation rule now lists the following: **When: Issue created, Issue Type equals Task, Then: Create 4 sub-tasks**.
14. On the right, name the automation **Create blog post subtasks**. Click **Turn it on** or **Turn on rule**. Your automation rule should be enabled.
15. Click on **Rule details** (under Automation) and explore the rule's details
16. In the sidebar, click **Back to project**.
17. Click on **Kanban board** if you're not already there.
18. Create an issue that triggers the rule:
    a. Click **Create** in the header at the top of the page.
    b. Ensure the project is **projectA**.
    c. Under **Issue Type**, select **Task**.
    d. Under **Summary**, enter **Create blog post 1**.
    e. Click **Create**.
19. View your **Kanban board**. You should see that the automation rule created four subtasks for your issue (you may need to refresh the browser window).
20. Click on a newly created subtask to view its details. Notice that the reporter is **Automation for Jira**. This is the "user" behind the scenes that executes automation rules. Later, we will modify the rule so that the reporter is the same as the parent task.
21. Back on the Kanban board, move the subtasks and parent task to the **Done** column.
22. Under **Project settings**, click **Automation**. (Or click the lightning bolt icon above the board and select **Manage automations**.)
23. Click on your **Create blog post subtasks** rule and select **Audit log**. You should see a recent entry stating that the rule was executed successfully. Click **Show more** to see details.
24. Click **Back to project** in the sidebar.

Congratulations, you have created an automation rule that creates subtasks when creating an issue. You have completed this lab.

Lab 3 - Create More Rules

Estimated time: 25 minutes

> ⓘ Note: This lab assumes that you have created an automation rule in the previous lab.

Exercise 1 - Create an automation rule that creates subtasks if the rule is manually executed.

> ⓘ Let's say that you only want to create subtasks when an issue is created in very specific circumstances. The rule that you created in the previous lab will trigger whenever any issue of type **Task** is created. We will modify that rule so that instead a user manually triggers the rule from a menu item in an issue's details.

1. Navigate to your **Create blog post subtasks** automation rule (under **Project settings > Automation**).
2. Select the **When: Issue created** trigger and click the **pencil** to the right of the **Issue created** heading to change the trigger.
   Note: If the pencil icon is not available, remove the trigger by clicking the trash icon.
3. Select **Manual trigger from issue**. Leave the dropdown as **All logged in users** (notice that you can limit access to this functionality if you want) and click **Save**.
4. When you created subtasks in the previous lab, the Reporter was set to “Automation for Jira”. Change the Reporter on each subtask to match the Reporter on the parent issue:
   a. Click the **Create 4 sub-tasks** action.
   b. Click **Add fields** next to the first subtask. Notice the **Create sub-task** action has changed to the **Create issue** action. This action allows you to set more field values.
   c. Click the **Choose field to set...** dropdown.
   d. Select the **Reporter** field.
   e. Click the new **Reporter** dropdown and select **Copy from trigger issue**. This means that the subtask will have the same Reporter as the issue that the rule was executed from (the trigger issue).
   f. Click **Save** or **Next**.
   g. (If desired) Repeat this process to modify the reporter of the remaining three subtasks.
5. Click **Publish changes** or **Update**.
6. Navigate to your **Kanban board**.
7. Click the **Create** button in the header at the top of the page and create an issue in projectA of type **Task** with a summary of **Create blog post 2**.
8. On the Kanban board, notice that the new issue does not have any subtasks. This is because we changed the automation rule to use a manual trigger.
9. Click on the new issue on the Kanban board to open the issue.

10. Next to workflow status, select **Actions** dropdown menu. This menu will show all automation rules for the project that have manual triggers.
11. Under **FROM AUTOMATION**, click **Create blog post subtasks**.
    - This is a powerful way for a Jira project administrator to easily extend the functionality of Jira in project-specific ways. The rules that they create only impact their project.
12. Close the issue and refresh the Kanban board. Verify that your **Create blog post 2** issue now has subtasks.
13. (optional) Create an issue of type **Story** with a summary of "Test story" and verify that manually executing the automation rule will result in the condition not passing and the action not executing. You can verify this under the **Audit** section for the rule.
14. On the Kanban board, move any issues in the **Backlog** column to the **Done** column.

Congratulations, you have created an automation rule that creates subtasks if the rule is manually executed.

Exercise 2 - Create an automation rule that auto-assigns high priority issues.

1. Navigate to **Automation** (under **Project settings**).
2. Click **Create rule**.
3. For the trigger, select **Issue transitioned**.
    - Leave the **From status** blank.
    - For the **To status**, select **Selected for Development**.
4. Click **Save** or **Next**.
    - Notice that the trigger will execute when the issue's status changes from any status to **Selected for Development**. This is when the issue should be auto-assigned (if necessary). This type of behavior can also be achieved by editing the workflow. Using Jira project automation, a Jira project administrator or other administrators can easily add the functionality without editing the workflow. By default, the rule will only affect this project.
5. Select **IF: Add a condition** and add an **Issue fields condition**. Under **Field**, select **Priority**. For **Condition**, select **greater than or equal to**. For **Value**, select **High**. Click **Save** or **Next**.
    - Because we selected the "greater than or equal to" condition, the condition will pass for High and Highest priority issues.
6. Add another **Issue fields condition**. Under **Field**, select **Assignee**. Under **Condition**, select **is empty**. Click **Save** or **Next**.
    - We add this second condition for cases where an assignee has already been specified for the issue. We would not want to automatically overwrite that assignee. Both conditions must pass for the action to be executed.
7. Click **THEN: Add an action** and add an **Assign issue** action. Under **Assign the issue to**, select **A user in a defined list**. Under **Method to choose assignee**, select **Balanced workload**. Leave the query in the **JQL to restrict issues** field as it is (though you could change it to something like **statusCategory = "In Progress"** so that it only counts issues

that team members have in the In Progress column). Under **User list**, enter **Alana Grant** and **Ryan Lee**. Click **Save** or **Next**. This automation rule will automatically assign high priority issues to either Alana Grant or Ryan Lee, depending on who has the least number of assigned issues that are not in the Done column.

8. Verify that under **Automation**, your automation rule now lists the following: **When: Issue transitioned to Selected for Development, Priority greater than or equal to High, Assignee is empty, Then: Assign the issue to Member of user list using balanced workload assignment**.
9. Name your automation rule **Assign high priority issues**. Click **Turn it on** or **Turn on rule**.
10. Navigate to your **Kanban board** and create an issue of type **Story** with a summary of **add item 1**. Assign a priority of **High**. After the issue is created, it should not be assigned (you should not see a user icon on the story's card on the board).
11. Drag the story to **Selected for development**. You should see either Alana Grant or Ryan Lee as the assignee (you may need to refresh the browser). If not, check the **Audit log** for the automation rule (under Project Settings > Automation) to see why the action didn't execute.
12. Create another **Story** with a summary of **add item 2** with a **High** priority and drag it to **Selected for development**. You should see that the issue is assigned to the other person in the user defined list (Alana Grant or Ryan Lee). This is because you selected **Balanced workload** under **Method to choose assignee**.
    - Notice that this automation rule does not cover the case where an issue is moved to the "Selected for development" column, and then later, the priority is changed to high. You could add another rule to trigger (on "Field value changed").
13. Move all issues to the **Done** column of the Kanban board.

Congratulations, you have created an automation rule that auto-assigns high priority issues.

Exercise 3 - Modify the rule to use an If / else block condition.

> ⓘ In this exercise you'll utilize If/else block which performs alternate actions based on whether certain conditions match or don't match. This powerful condition allows you to add as many if/else conditions as you want.
> This rule will run when an issue is transitioned and assign to different users based on certain conditions using the if/else block.

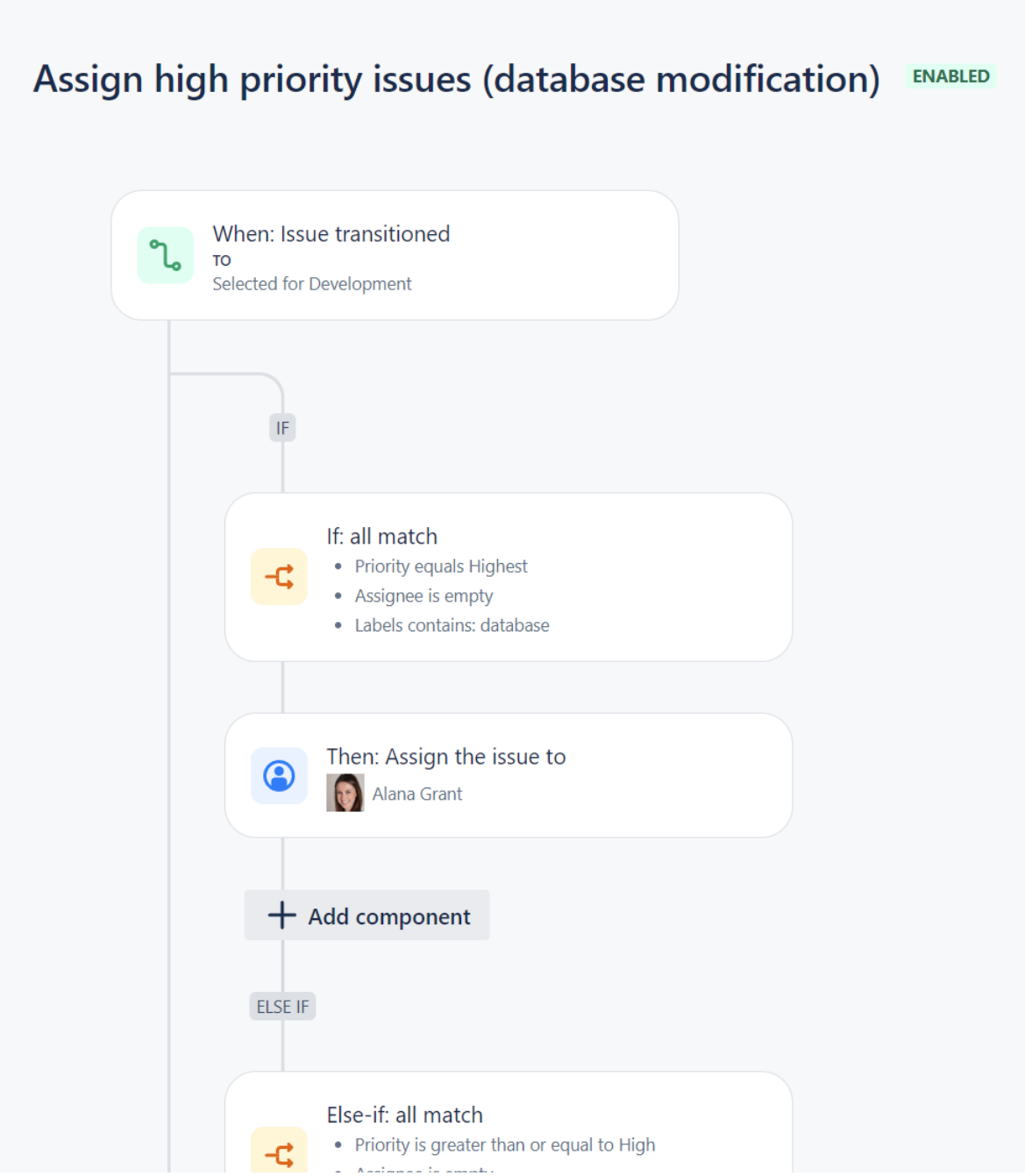
1. Add a **database** label to one of the issues in your project. This is so you can use the label in your automation rule.
    a. Select one of the issues you already created and have moved to **Done**.
    b. In the **Details** section, click **None** to the right of Labels.
    c. Type **database**, and select **database (new label)** in the dropdown.
    d. Close the issue.
2. Use the **Assign high priority issues** rule to create a new rule. The requirements for this rule are:

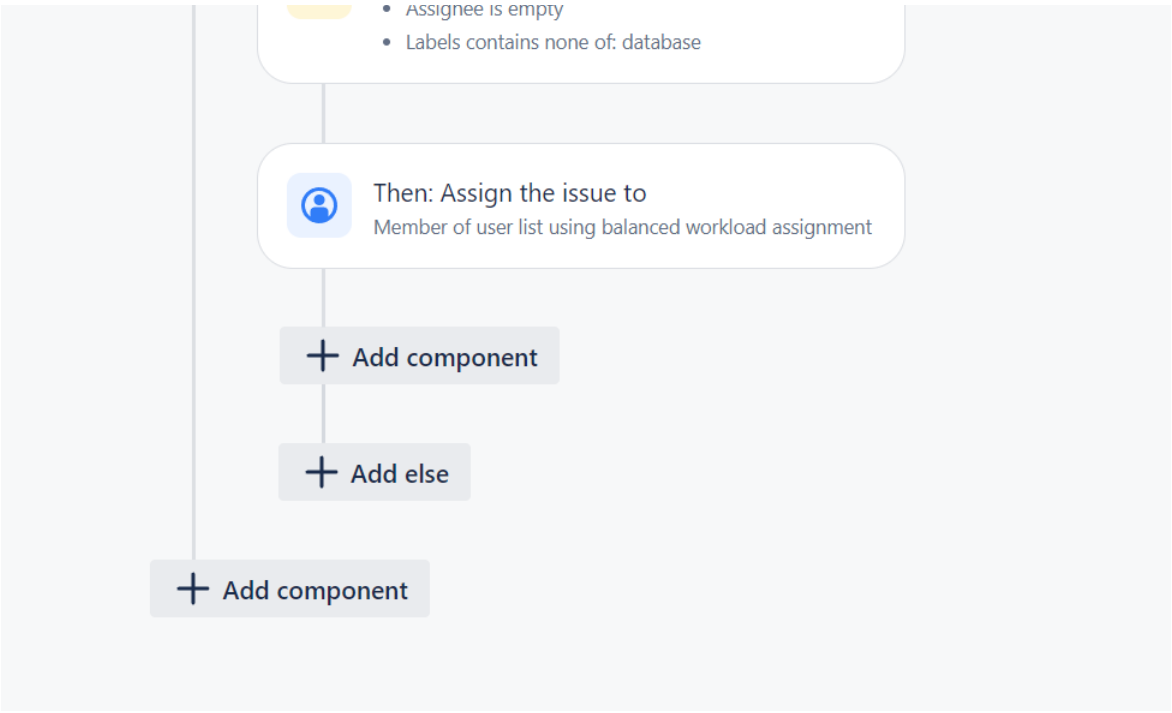
- **Alana Grant** is assigned any unassigned issues that have a priority of **Highest** and a label containing **database**.
- All other unassigned issues with a priority of **High** or greater will be assigned to the team member with the least number of open issues. (This was part of the existing rule.) Tip: Use the rule shown near the end of the module as a guide.

3. Navigate to the Automation page.
4. Click **Create rule**.
    a. For the trigger, select **Issue transitioned**. Leave the **From status** blank. For the **To status**, select **Selected for Development**. Click **Save** or **Next**.
    b. Select **IF: Add a condition** and add an **IF, ELSE: add condition options** block. Create conditions for **Alana Grant** to be assigned unassigned issues with a priority of **Highest** and a label containing **database** here.
        i. Click **Add conditions...**
        ii. Select **Issue fields condition**. Under **Field**, select **Priority**. For **Condition**, select **equals**. For **Value**, select **Highest**.
        iii. Click **Add conditions...**
        iv. Select **Issue fields condition**. Under **Field**, select **Assignee**. For **Condition**, select **is empty**.
        v. Click **Add conditions...** again.
        vi. Select **Issue fields condition**. Under **Field**, select **Labels**. For **Condition**, select **contains any of**. For **Value**, select **database** (this should be available if you created it at the beginning of the exercise in an existing issue).
        vii. Click **Save** or **Next**.
        viii. Click **THEN: Add an action**.
        ix. Select **Assign issue.**
        x. Assign the issue to: **Specify user.**
        xi. Select a user: **Alana Grant.**
        xii. Click **Save** or **Next**.
    c. Select **Add else** located at the bottom of your Rule details. Create conditions for any other unassigned issues with a priority of **High** or greater excluding the label **database** to be assigned to the team member with the least amount of open issues here.
        i. Click **Add conditions...**
        ii. Select **Issue fields condition**. Under **Field**, select **Priority**. For **Condition**, select **greater than or equal to**. For **Value**, select **High**.
        iii. Click **Add conditions...**
        iv. Select **Issue fields condition**. Under **Field**, select **Assignee**. For **Condition**, select **is empty**.
        v. Click **Add conditions...** again.
        vi. Select **Issue fields condition**. Under **Field**, select **Labels**. For **Condition**, select **contains none of**. For **Value**, select **database**.
        vii. Click **Save** or **Next**.
        viii. Click **THEN: Add an action**.
        ix. Select **Assign issue.**

x. Assign the issue to: **A user in a defined list.**
xi. Method to choose assignee: **Balanced workload.**
xii. Leave the **JQL to restrict issues** as default.
xiii. Under User list add **Alana Grant** and **Ryan Lee.**
xiv. Click **Save** or **Next**.

d. Name your automation rule **Assign high priority issues (database modification)**.
e. Your issue rule details should look like the picture below.





   f. Click **Turn it on** or **Turn on rule**.
5. Once you see the green tick and message **'Your automation has been turned on'** click **Return to list** or **Return to rules** in the top right.
6. You should now see the issue rule you have just created along with the original **Assign high priority issues** rule you created in the previous exercise. To the right of the original rule click the green tick under the **Enabled** column to turn the rule off as we no longer need it.
7. Test your rule to make sure that **Highest** priority issues with a label of **database** get assigned to **Alana Grant**. If necessary, use the Audit log details to help troubleshoot.
   a. Click **Back to project** in the sidebar and make sure you are on the **Kanban board**.
   b. Click **Create** in the top menu.
      i. Issue Type: **Story** (we didn't define an issue type in our rule so this won't affect your rule).
      ii. Summary: **Test database rule 1**
      iii. Assignee: **Unassigned**
      iv. Priority: **Highest**
      v. Labels: **database**
   c. Click **Create**.
   d. Move to **SELECTED FOR DEVELOPMENT** and refresh the page to see who is assigned to the issue.
   e. If **Alana Grant** isn't assigned to the issue go back and see what has been missed using the **Audit log** details to help troubleshoot.
8. (OPTIONAL) Test your new rule on an unassigned issue without the database label. If you have left your Kanban board return first.

a. Click **Create** in the top menu.
    i. Issue Type: **Story**.
    ii. Summary: **Test database rule 2**
    iii. Assignee: **Unassigned**
    iv. Priority: **High**
    v. Labels: ***Leave blank***
b. Click **Create**.
c. Move to **SELECTED FOR DEVELOPMENT** and refresh the page to see who is assigned to the issue.
d. If **Ryan Lee** isn't assigned to the issue go back and see what has been missed using the **Audit log** details to help troubleshoot.

Congratulations, you have completed this lab.

Lab 4 - Administration

Estimated time: 20 minutes

Exercises:

1. Create a kanban project for testing rules.
2. Create a rule that writes to the audit log.
3. Limit groups that can trigger a rule.
4. Copy a rule to another project.
5. Explore global administration.

Exercise 1 - Create a kanban project for testing rules.

1. (If you are not still logged in as **admin**) Navigate to the URL of your Cloud site and log in to Jira as **admin**.
    - Note: The **admin** user is a Jira administrator. Jira administrators can create company-managed (formerly classic) projects.
2. If you see onboarding screens such as "Invite your team" and "Help us set up your Jira" screens, click **Skip**.
3. In the top navigation, click **Projects > Create project**.
4. In the “Project templates” screen, select **Software development** located in the sidebar. Click the **Kanban** template. Click **Use template**.
5. Click **Select a company-managed project**. If you are unable to create a company-managed project, verify that you are logged in as **admin**.
6. In the **Add project details** screen, enter **projectATest** for the project name. You can leave the default **Key** as is. Verify that the template is **Kanban**. Click **Next**.
7. On the following screen, choose **Skip** or **Continue without connecting**. You should see the Kanban board for your new project.

Congratulations, you have created a kanban project for testing rules.

Exercise 2 - Create a rule that writes to the audit log.

> ⓘ Writing to the audit log is a safe thing to do as you are writing and troubleshooting rules.

1. In the sidebar, click **Project settings > Automation**.
2. Click the **Rules** tab under **Automation**.
3. Click **Create rule** in the upper right. If **Create rule** isn’t visible you may need to click **Return to list** first.
4. Click on the **Scheduled** trigger. Set the rule to run every **1 Days**. We will see that we can run scheduled rules at any time using the automation interface. Click **Save** or **Next**.
5. Add a **THEN: Add an action** and select **Log action** action (under Advanced). In the **Log message** text box, enter **Hello world**. Click **Save** or **Next**.

6. Verify that under **Automation**, your automation rule now lists the following: **When: Scheduled, Then: Add value to the audit Log**.
7. Name the automation **Write to audit log**. Click **Turn it on** or **Turn on rule**. Your automation rule should be enabled.
8. Click on **Rule details** (under Automation) and explore the rule's details. Click the ellipsis (**...**) near the top and notice the **Run rule** button. This is a handy way to run scheduled rules at any time. This is a good trick that may be helpful when creating rules.
9. Click **Run rule**.
10. Click **Audit log** (under Automation) and click **Show more** on your recent entry. You should see **Hello world** in the log.

Congratulations, you have created a rule that writes to the audit log.

Exercise 3 - Limit groups that can trigger a rule.

ⓘ We will limit the scope of the rule so that only Jira administrators can trigger it.

1. Modify the **Write to audit log** rule. Click on the **Scheduled** trigger and click the **pencil icon** (edit) to change it.
   Note: If the pencil icon is not available, remove the trigger by clicking the '**x**' icon.
2. Add a **Manual trigger**. Under **Groups that can run trigger**, select **jira-administrators** (or **jira-admins-<site-name>**). This means that the logged in user must belong to this group in order to execute the rule. This is an easy way to limit the scope of some rules. Click **Save** or **Next**.
3. **Update** or **Publish changes** to the rule.
4. Manual rules are triggered from an issue's details. Create a new issue, of any issue type, in your project named **add item 1**, then open it. Select **Actions** dropdown menu and run the **Write to audit log** rule. Verify that **Hello world** was written to the audit log.
5. (optional) Click on your user icon in the upper right and select **Log out**. Log in as **Alana Grant** (login instructions should have been provided to you at the beginning of the course). Verify that Alana Grant can not trigger this rule. This is because we have set this rule to only trigger if the logged in user is a Jira administrator. Log back in as **admin**.

Congratulations, you have limited the groups that can trigger a rule.

Exercise 4 - Copy a rule to another project.

ⓘ Let's assume that we are happy with a rule and want copy it to another project for additional modifications. This leaves the original rule available as an independent rule.

1. Navigate to **Project settings > Automation** for your **projectATest** project.

2. With your project's rules listed, click the **Global administration** link (this link is available to all Jira administrators, but executing global and multi-project rules requires a premium cloud subscription).
3. Click on **All rules**. This displays all global (multi-project) and project rules in your cloud site.
4. Click on the **Write to audit log** rule to view its details.
5. Select the **Scope** dropdown and view your options.
    - Single project (the current setting) - The rule can only be executed in a single project.
    - All projects - Converts this rule to a global rule.
    - Multiple projects - Allows you to choose which projects can execute the rule. All of these projects share the same rule (changes to the rule affects all projects).
    - Project type - The rule applies to all projects of a certain type (for example Jira Software or Jira Service Desk).
6. Click **Return to list** or **Return to rules** to cancel and avoid changing the original rule. We will copy the rule instead.
7. Click **All rules** again, if needed, to view the list of rules on your cloud site.
8. Hover your mouse over the **Write to audit log** rule and click the more icon (**...**). Notice that you can Copy, Remove or Export the rule. Exporting the rule allows you to import it on another cloud site. Select **Copy**.
9. Change the name of your copied rule to **Write to audit log projectA**. Change the **Projects** dropdown to **projectA**. Click **Save**. Click **Publish copy** or **Turn on rule**.
10. Navigate to **projectA**, open an issue, and verify that you can successfully execute the **Write to audit log projectA** rule. This rule is independent of the original **Write to audit log** rule.

Congratulations, you have copied a rule to another project.

Exercise 5 - Explore global administration.

1. While in any project navigate to **Project settings > Automation** and click **Global administration**. Notice that this takes you to the **Rules** tab in Jira **System Global automation** page. You could also reach this tab by clicking the gear icon in the upper right and selecting **Jira Settings > System**.
2. Under Automation, click **Audit log**. This displays a single audit log for all rule executions on your cloud site.
3. Click the more icon (...) in the upper right and select **View performance insights**. This shows the execution details for the cloud site's rules. This helps troubleshoot any automation-related performance issues that rules may be causing. Click **Return to list** to navigate back to the previous screen.

4. Click the more icon (...) in the upper right and select **Transfer user**. This allows you to swap users related to automation rules. This is especially helpful if someone leaves the team. Click **Cancel**.
5. Click the more icon (...) in the upper right and select **Global configuration**. Notice that by default, all project administrators can manage rules. If you want to lock down project automation rule management, you could deselect the checkbox. Another option is to create a Jira group named something like automation-managers and only allow users in that group to manage project automation rules. Click **Return to list**.
6. Click the more icon (...) in the upper right one last time and notice that you can also **Import** and **Export** rules. Click out of the drop down area to close it.
7. Under **Automation**, click the **Usage** tab. This show your monthly usage of automation rules. This is useful to monitor your site if you have a plan with usage limits.

Congratulations, you have explored global administration and completed this lab.

Lab 5 - Smart Values

Estimated time: 10 minutes

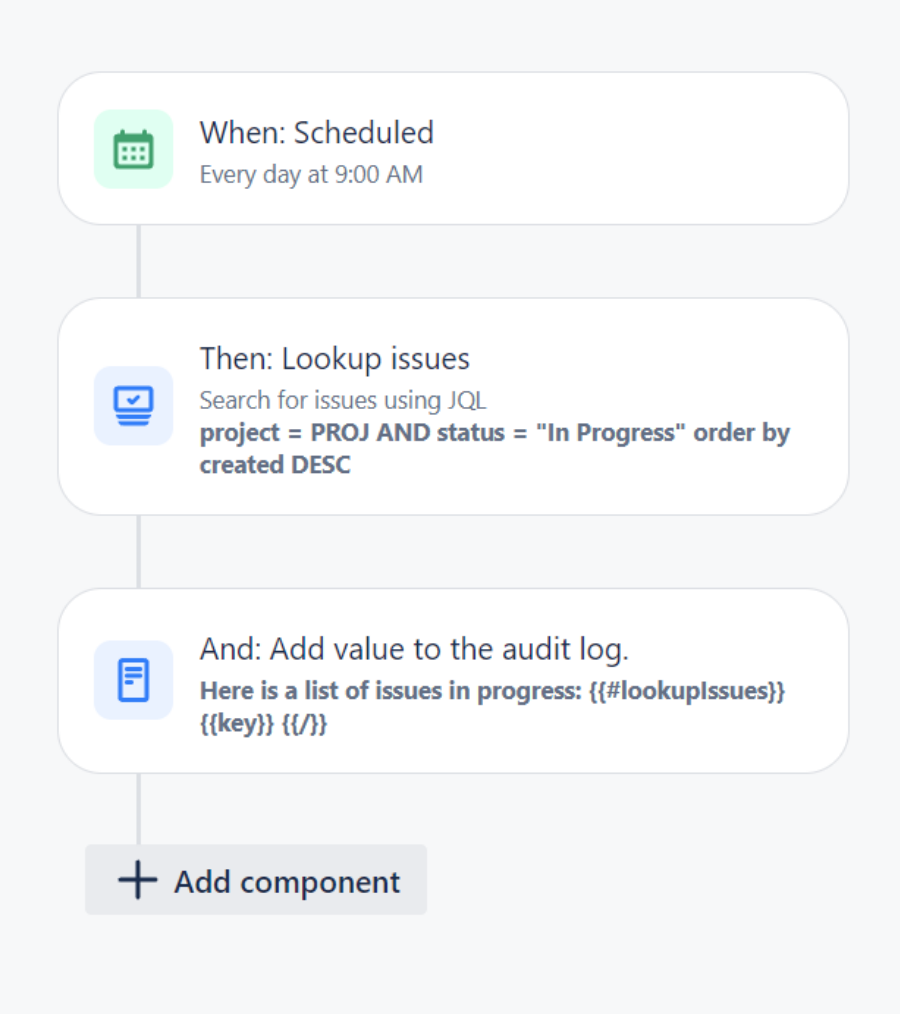
Exercises:

1. Create a rule that lists the keys for all in progress issues in a project

Exercise 1 - Create a rule that lists the keys for all open issues in a project

> Tip: If you need more help than what is provided below, use Module 5 of the student guide for more information.

1. On your **projectA** Kanban board, move some issues to the **In Progress** column.
2. Create a JQL query that searches for all issues in **projectA** that have a **Status Category** of **In Progress**:
    a. Click the **Search** box at the top right of your screen and select **View all issues**.
    b. In basic search, search for issues of **projectA** with a **Status Category** of **In Progress** (if options aren't visible you may need to use the +More link). Verify that your search results contain the issues from the **In Progress** column of your Kanban board.
    c. Click **Switch to JQL** and **copy** the JQL to your clipboard.
3. Create an automation rule in **projectA** named **Today's issues in progress**:
    a. Navigate to the Automation page (**Project** drop down > **ProjectA** > Automation shortcut (**lightening bolt** icon) > Manage automations > Create rule)
    b. Click **Scheduled** trigger and set to run every day. Click **Save** or **Next**.
    c. Add a **THEN: Add an action** and select the **Lookup issues** action that uses the JQL copied to your clipboard above. Click **Save** or **Next**.
    d. Add a **THEN: Add an action** and select the **Log action** action to list the issue keys of the JQL pasted in the **Lookup issues** action.
        - You could use this as the message body: **Here is a list of issues in progress: {{#lookupIssues}} {{key}} {{/}}**
        - Note: You could replace **{{key}}** above with **{{url}}** to log URLs of the issues rather than their keys.
        - Note: You could also send this list of issues to your personal email address if you would like using the **Send email** action.
    e. Name your rule **Today's issues in progress** and click **Turn it on** or **Turn on rule**.
    f. Your rule should look as the picture below.



4. Click the ellipsis (**...**) and then select the **Run rule** button at the top of your rule to trigger the rule. This button appears for rules with a **Scheduled** trigger. Verify that your Audit log contains the list of **In progress** issue keys.
5. (optional) Change your **Audit log** action to show other information, such as the issues' **Summary**.
6. Disable your rule so that it doesn't run tomorrow.

Congratulations, you have created a rule that lists all in progress issues and completed this lab.

Lab 6 - Advanced Rules

Estimated time: 15 minutes
Exercises:

1. Create an automation rule that copies a comment from a child issue to the parent epic
2. (Optional) Create and use a rule with an incoming webhook trigger

Exercise 1 - Create an automation rule that copies a comment from a child issue to the parent epic

> ⓘ If any issue of an epic is commented on, the comment is copied to the epic's comments. This creates a convenient compiled list of comments related to the epic.

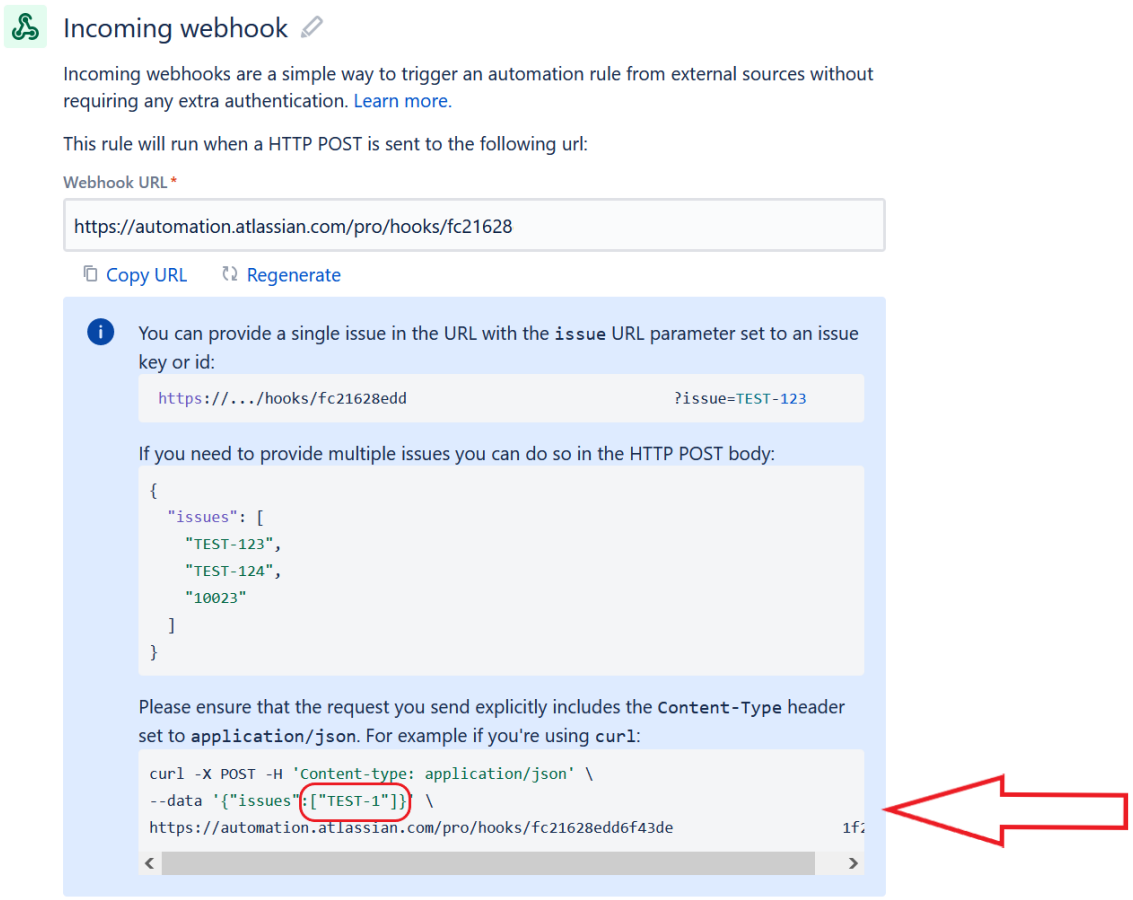
1. Navigate to **ProjectA** and go to **Automation** (under **Project settings**).
2. Click **Create rule**.
3. For the trigger, select **Issue commented**. Click **Save** or **Next**.
4. Add a **IF: Add a condition** and select **JQL condition**. Enter `"Epic Link" is not EMPTY` as the JQL. Click **Validate query** to make sure the syntax is valid. This rule will only continue if the triggered issue has an associated epic. Click **Save** or **Next**.
5. Click **FOR EACH: Create a branch** and add a **Branch rule / related issues**. Under **Type of related issues**, select **Epic (parent)**. Click **Save** or **Next**.
6. Add a **THEN: Add an action** and select **Comment on issue** action. For the comment, enter **A comment was added on issue: {{triggerIssue.key}} - {{triggerIssue.summary}} Text of comment: {{triggerIssue.comment.body}}**. This uses smart values to include relevant data in the comment. Click **Save** or **Next**.
7. Verify that under **Automation**, your automation rule now lists the following: **When: Rule is triggered on** All comments**, If: Issue matches JQL, For Epic (parent), Then: Add comment to issue**.
8. Name your automation rule **Copy comment to parent**. Click **Turn it on** or **Turn on rule**.
9. Navigate to your **Kanban board** and create an issue of type **Epic** (using the **Create** button in the header) with the name **Feature A** and a summary of **add feature A**.
10. Click the **add feature A** issue to open its details.
11. Under the summary, click **Add a child issue**. Create an issue of type **Story** with a summary of **add item A1**.
12. Open the **add item A1** story and add a comment: **This comment should be added to the parent epic.**
13. Open the **add feature A** epic and verify that the comment from the epic's story was automatically copied to the epic, along with additional information from the smart values. If you don't see the comment, you may need to refresh the page.
14. Move all issues on the Kanban board to the **Done** column.

Congratulations, you have created an automation rule that copies a comment from a child issue to the parent epic.

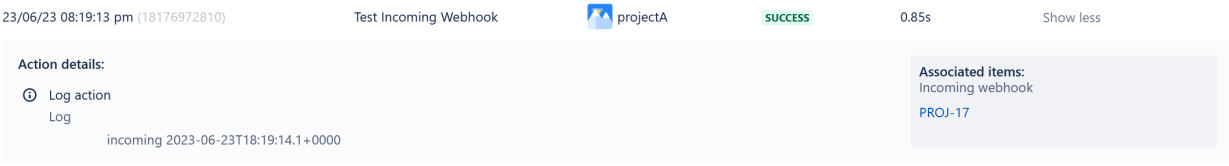
Exercise 2 - (Optional) Create and use a rule with an incoming webhook trigger

To be able to complete this exercise, you should be familiar with using a terminal or command line interface.
In this exercise you will create an automation rule named **Test incoming webhook** that logs an entry to the Audit log when an incoming webhook is called.
You will also be using the **curl** command, which you may need to install first depending on the operating system you are using.

1. Create a rule with an **Incoming webhook** trigger.
2. Add a **THEN: Add an action** and select **Log action** that prints “incoming webhook {{now}}” to the Audit log. You can also edit time formatting and use {{now.shortDateTime}} instead.
3. Name the rule **Test incoming webhook**.
4. Copy the **curl** command provided in the **Incoming webhook** trigger information to your clipboard. You might need to drag your mouse below the bottom of the textbox to make sure that you copy the entire URL.
    a. Don’t forget to replace the TEST-1 issue key in the command with an actual issue key from your Jira instance:



5. Open a terminal / command line interface and execute the **curl** command with the modified issue key.
6. View your rule's Audit log to verify that the incoming webhook triggered the automation rule. It should look like this:



7. (optional) Modify the log entry to display the data from the curl command.

Congratulations, you have completed this lab.

Lab 7 - Jira Service Management Automation

Estimated time: 15 minutes

Exercises:

1. View Jira Service Management automation.
2. Create a new Jira Service Management automation rule.

Exercise 1 - View Jira Service Management automation

> ⓘ Jira Service Management comes with unique automation features. Here we'll view its unique template rules, trigger, and actions.

1. Log in to your cloud site as **admin**.
2. Create a Jira Service Management project:
    a. In the top navigation, click **Projects** then select **Create project**.
    b. Click **Service management** in the left sidebar.
    c. Select **IT service management** the click **Use template**.
    d. For the project name, enter **IT Service Management**.
    e. (If necessary) Enter **ISP** for the Key.
    f. Select **Information Technology (IT)** for **Team type** or **What team is this for?**
    g. Select **Restricted** for **Channel access**.
    h. Click **Create project**.
3. In the sidebar, click **Project settings**.
4. Click **Automation**.
5. (If necessary) Select the **Automation** tab (not the Legacy automation tab).
6. Explore the **Library** (or **Templates**) tab.
    - **Note**: Here you see all the preset rules specific to a Jira Service Management project. You can use these as is by simply enabling them or editing them to suit your needs.
7. Click the **Rules** tab.
8. View triggers and actions unique to Jira Service Management:
    a. Click **Create rule** (top right).
    b. If you start the tour, click **Skip tour**.
    c. Scroll down to the **Issue triggers** section and click **SLA threshold breached**.
    d. Click the **SLA** dropdown and select any SLA.
        - **Note**: This trigger is unique to Jira Service Management and can be used to trigger automation when an SLA is breached, is about to breach, or has breached in the past.
    e. Click **Save** or **Next**.
    f. Click **THEN: Add an action**.
    g. Scroll down to the **Jira Service Management** actions.

- **Note**: These actions are unique to Jira Service Management and can be used to perform actions such as adding a Jira Service Management customer to the project, approving or declining a request, or creating a new request in another project.

h. Scroll back up and click **Return to list** or **Return to rules**, then **OK** to discard the new rule.

Exercise 2 - Create a new Jira Service Management automation rule

ⓘ Some service requests require approval from a manager, for example requesting a new mobile device. Here we'll create a rule that automatically approves new mobile device requests if the user is an administrator.

1. Create a new automation rule:
   a. Click **Create rule**.
   b. For the trigger, click **Issue created** and then click **Save** or **Next**.
   c. Create a condition that passes if this is a new mobile device request:
      i. Click **IF: Add a condition**.
      ii. Click **Issue fields condition**.
      iii. For **Field**, select **Request Type** (near the end).
      iv. For **Condition**, select **equals**.
      v. For **Value**, select **New mobile device**. This is a request type that requires approval.
      vi. Click **Save** or **Next**.
   d. Create a condition that passes if the reporter is an administrator:
      i. Click **IF: Add a condition**.
      ii. Click **User condition**.
      iii. For **User**, select **Reporter**.
      iv. For **Check to perform**, select **is in group**.
      v. For **Criteria**, select **jira-admins-<sitename>** (e.g. jira-admins-jir-92)
      vi. Click **Save** or **Next**.
   e. Create an action that approves the request:
      i. Click **THEN: Add an action**.
      ii. Scroll down and, in the Jira Service Management section, select **Approve/ Decline request**.
      iii. For **Approval decision**, select **Approve**.
      iv. Click **Save** or **Next**.
   f. Verify your automation rule now lists the following:
      - **When: Issue created**
      - **(If:) Request Type equals**
        New mobile device

- **(And:) Reporter is in**
  jira-admins-<sitename>
- **Then: Approve/Decline request**
  Approve

g. On the right, name your rule **Approve mobile devices for admins** and click **Turn it on** or **Turn on rule**.

2. Create a new request:
   a. (If necessary) Click **Back to project settings**.
   b. In the **Project settings** sidebar, click **Portal settings**.
   c. Copy the **Portal URL** and open the link in a new browser tab. This takes you to the customer portal.
   d. Click **Computers**.
   e. Click **New mobile device**.
   f. For Summary, enter **I need a new phone**.
   g. For who is your manager? select **Alana Grant**.
   h. Click **Send**. In the issue, there's an automatic response that indicates approval is needed.
3. Verify the request was approved:
   a. Return to the **IT Service Management** project browser tab.
   b. (If necessary) Return to the project and click **Queues** in the sidebar.
   c. Click the new request's key or summary to open the request.
      i. If you don't see the issue, refresh the window.
   d. Look at the **Activity** and you should see **Status approved**.
   e. Click the status dropdown in the upper right (**Waiting for support**) and select **View workflow**.
   f. You should see that the issue has passed the WAITING FOR APPROVAL status. Close the workflow window.
4. View the audit log:
   a. Click **Project settings**.
   b. Click **Automation**.
   c. Click the **Audit log** tab in the upper left.
   d. In the top row for your new automation rule (Approve mobile devices for admins), you should see a status of SUCCESS. Click **Show more**. Note that the final action detail shows the approval has been completed.
5. Optionally, test the rule as a user who is not an administrator:
   a. In **Project settings**, click **Customer permissions**.
   b. Under Customer sharing, select **Customers can search for other customers within their project or organizations**, if necessary and click **Save**. This will allow the user to select their manager in the customer portal.
   c. In Project settings, click **Portal settings** and copy the **Portal URL**. You need this to submit a request as another user.
   d. You can log in as **Ryan Lee** in a couple of ways, choose one:
      i. Open an incognito or private window in your current browser.
      ii. Open a separate browser if you have another one installed.

iii. Log out as **admin** and log back in again.

e. Open a new browser tab, paste and go to the portal URL.
f. In the customer portal, (if necessary) select **IT Service Management** then **Computers** to create a new request for a mobile device, specifying **Alana Grant** as the manager.
g. As **admin**, view the new request and confirm the request was not automatically approved. The status should be **Waiting for approval**.

Congratulations, you have created an automation rule that automatically approves administrator requests for new mobile devices. You have completed this lab.