

Log work

This post function automatically **creates worklogs** for one or multiple issues.



Configuration

Target issue

Select the issue(s) to log work on. The following options are available:

Option	Description
Current issue	Work will be logged on the current issue .
Parent issue	Work will be logged on the current issue's parent .
Sub-tasks	Work will be logged on the current issue's sub-tasks . If you choose this option you can further filter your selection (e.g. by issue type or status). Only useful for standard issues .
Sibling sub-tasks	Work will be logged on the current issue's sibling sub-tasks . Only useful if the current issue is a sub-task .
Linked issues	Work will be logged on linked issues . If you choose this option you can further filter your selection (e.g. by issue type or status).
Linked epic	Work will be logged on the related epic .
Issues under epic	Work will be logged on all issues related to the same epic. If you choose this option you can further filter your selection (e.g. by issue type or status).
Sibling issues under epic	Work will be logged on all sibling issues related to the same epic. If you choose this option you can further filter your selection (e.g. by issue type or status).
Set target issue manually (parser expression)	Define the issues that work will be logged on by defining an expression . The JQL mode and Issue list mode are available. <div>The expression must return either a valid JQL expression or an issue list with issue keys.</div>

Time

You can either enter the **number of minutes** or calculate the value in the [Numeric mode](#). The result should be the number of minutes to be logged.

Starting date-time

Specify the starting date-time for the time log. Since a timestamp is always represented by a number you must specify the expression in the [Numeric mode](#).

If leaving this field blank, the **current date-time** is used.

Work description

This parameter adds a standard description to your work log. You can either enter plain text and optionally use [field codes](#) to insert field values using the [mixed parsing mode](#).

Additional options

Remaining estimate

You can decide how to proceed with the remaining estimate of the issues. You have the two following options:

Option	Description
Auto adjustment	The issues' remaining estimate will be reduced by the amount of work logged.
Retain remaining estimate	The issues' remaining estimate will not be updated.

Conditional execution

You can **optionally** specify a [logical expression](#) to define the circumstances (or conditions) under which the post function should be executed.

The result of the logical expression must return a boolean value of either:

- **true** the post function will be executed
- **false** the post function will **not** be executed

Using the **conditional operator**, even complex or multi-layered conditions can be constructed.

Make sure to learn more about defining logical expressions and browse through the various **examples** here: [Logical mode](#)

Run as

Select which **user** will be used to execute this post function. By default this parameter is set to the **current user**. You can also use field codes to run the function as a dynamic user (e.g. current assignee).

Make sure that the user running the post function has all the **relevant permissions** to perform the actions defined in the configuration (e.g. "Update Issues")!

If you want to keep track the actions being performed automatically, we suggest to create a **dedicated JWT account**, granted all relevant **permissions**, and use it in the Run as parameter to identify which changes have been made with JWT.



Use cases and examples

Use case	JWT feature	Workflow function	Parser functions	Label
Automatically log work spent in a specific status		Log work	timeDifference()	
Log absence time on another issue		Log work	stringToDate()	
Add percentaged profit margin when closing issue		Log work		

If you still have questions, feel free to refer to our [support team](#).