Restrict issue creation per issue type and project role

On this page

Configuration | Screenshots | Related use cases

Use case

In Jira it is not possible to restrict the creation of issues with specific issue types to certain user/ user groups/ project roles.

In the following use case, we restrict the issue creation depending on the selected **issue type.** Each issue type can only be created by a **certain project role**.

In this use case we assume that **several issue types are using the same workflow**.

If you have one workflow for each issue type, check out the **corresponding use case**.

Configuration

1 Add the validator to the **create transition** of the desired workflow.

2

Expression

Enter the following expression:

```
(%{issue.issueType} = "Bug" implies (isInRole(%{system.currentUser}, "Developers") or isInRole
(%{system.currentUser}, "Software Testers"))) and
(%{issue.issueType} = "New Feature" implies isInRole(%{system.currentUser}, "Product
Managers"))
```

Adapt the expression to your needs!

Above expression explained:

Selected issue type for new issue	Explanation
Bug	Only if the current user is in role **Developers** or **Software Testers**, the issue can be created.
New Feauture	Only if the current user is in role **Product Managers**, the issue can be created.
Any other issue type	Can be created by any user independent of any project role.

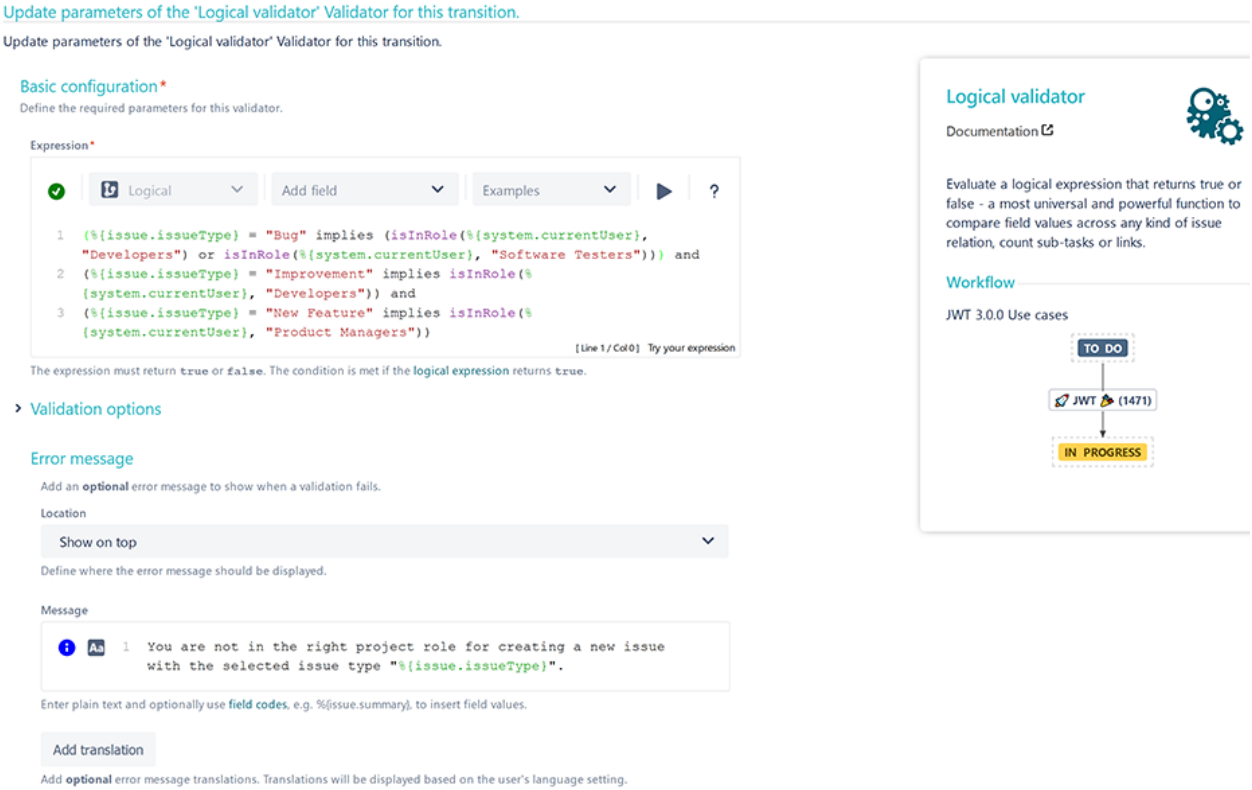
3

Error message

Enter the following expression in **basic text** mode:

```
You are not in the right project role for creating a new issue with the selected issue type "%{issue.issueType}".
```

Screenshots



Related use cases

Use case	JWT feature	Workflow function	Parser functions	Label
Block an Epic's transition depending on linked issues status and due date		Logical validator Logical condition	count() filterByPredicate() linkedIssues()	
Block an Epic's transition until all the issues under that Epic are resolved		Logical validator Logical condition	count() issuesUnderEpic() filterByResolution()	

Block a transition until all sub-tasks have certain fields populated		Logical validator Logical condition	count() filterByPredicate() subtasks()	STAFF PICK
Ensure that all issues linked with a certain issue link type have "Due Date" field set		Logical validator Logical condition	count() linkedIssues() fieldValue()	
Ensure that an issue has at least one attachment		Logical validator Logical condition		STAFF PICK
Evaluate Assets objects		Logical validator Logical condition	findPattern() findReplaceAll() replaceAll() toStringList()	
Halt a transition if an element is not contained in a list		Logical validator Logical condition	toStringList()	
Hide transition to a previous status		Logical condition		
Make "Time spent" field required		Logical validator		
Prevent a closed issue from being reopened after resting 1 week closed		Logical validator Logical condition	isInRole()	
Prevent a transition depending of the number of components		Logical validator Logical condition	numberOfSelectedItems()	
Prevent external users from creating issues		Logical validator	matches() toString() textOnStringList() toStringList() userDisplayName()	
Prevent issue creation if description contains less than 100 characters		Logical validator	replaceAll() length()	
Prevent issue creation with the same field value (Boolean)		Logical validator Validation based on JQL query	count() issuesFromJQL()	
Prevent setting due dates outside business hours		Logical validator Logical condition	withinCalendar()	STAFF PICK
Prevent the creation of sub-tasks unless the parent issue is in a certain status		Logical validator		

Reject duplicated file names in attachments		Logical validator Logical condition	count() toStringList() distinct()
Restrict issue creation per issue type and project role		Logical validator	isInRole()
Set a condition in a global transition which only applies in a certain status		Logical condition	
Validate a Select List (cascading) custom field		Logical validator Logical condition	
Validate only issue links created in transition screen		Logical validator	count() transitionLinkedIssues() filterByProject() filterByIssueType()
Validation based on the value of a date type project property		Logical validator Logical condition	stringToDate() projectProperty()

If you still have questions, feel free to refer to our support team.