## **JIRA Checklist Plugin Comparison**

(1)

Warning: stale info!

This comparison was done in 2016. The plugin landscape has evolved since then, and I have not had time to update this comparison.

Plan Sprint

CodeWrite docs

Sometimes in JIRA you want a checklist, or todo list for your issue, just like Confluence's Task List. What are your options?

1. You could model your todo items as sub-tasks. Advantages: todo items can have assignees, due dates, and a fancy workflow. Disadvantage: visual clutter, difficult to use, and you don't *need* the full power for JIRA for something this simple.

you don't *need* the full power for JIRA for something this simple.

2. You might try to use JIRA's built-in 'Checkboxes' field type for a 'Todo' field. This does not work. With JIRA checkboxes, only checked items show. Unchecked items are considered to be "irrelevant to this issue", and so don't display except when editing. A todo list with no completed items wouldn't show up at all. In a todo list, unchecked means "task is not yet completed", which makes it just as (or more) important than checked items.

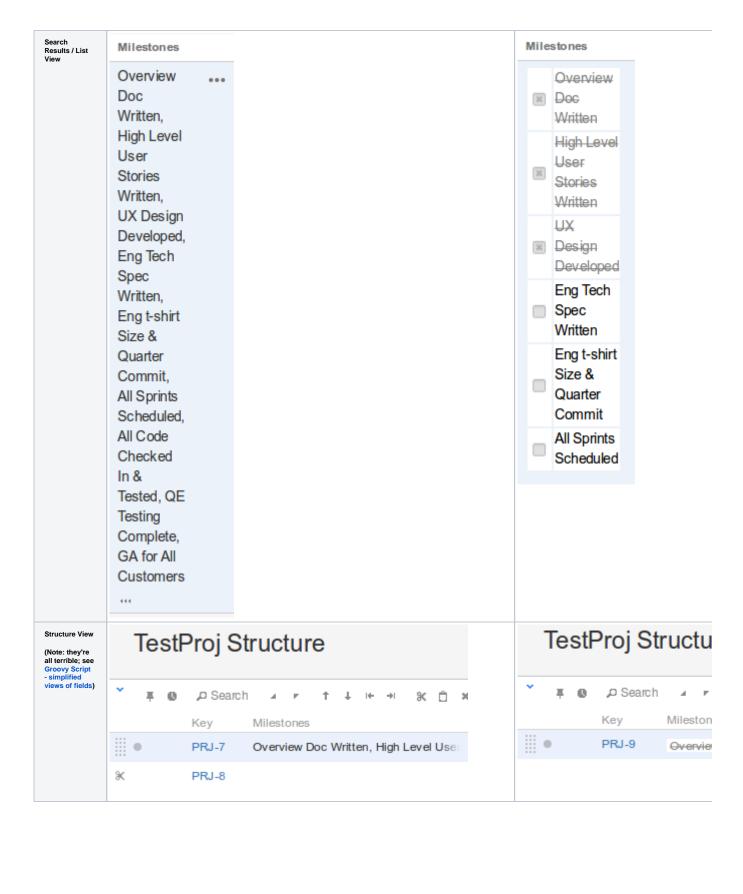
3. Consider the various JIRA checklist / todo list plugins.

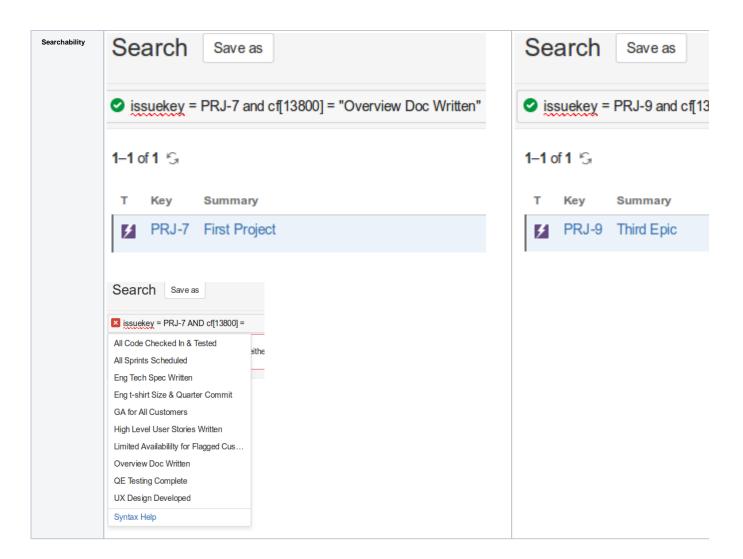
Here we review a handful of checklist / todo list plugins available (in February 2016), seeing how they stack up to each other and the built-in Checkboxes field.

	Built-in JIRA Checkboxes	My Todo (previously ToDo list custom field)
Price (Oct '20)	n/a	10 / 85 / 150 / 310 / 650 / 900 / 2200 / 2700 / 3500
Price (Feb '16)	n/a	\$0

	.IIRA only displays the v	alues of checkboxes that have been ticked.	N 4714	
Issue View	Milestones:	alues of checkboxes that have been ticked.  Overview Doc Written, High Level User Stories Written, UX Design Developed	Milestones:	M C
				×
				M L
				_ E
				Е
				_ A
All Checkboxes	Milestones:	Overview Doc Written, High Level User Stories Written, UX Design Developed, Eng Tech Spec Written, Eng t-shirt Size & Quarter Commit,	Milestones:	1 1
Ticked		All Sprints Scheduled, All Code Checked In & Tested, QE Testing Complete, GA for All Customers		
				M 1
				× -
				36

Edit View	Milestones	■ Overview Doc Written	Milestones: Milestones		
		High Level User Stories Written		×	Overview Do
		UX Design Developed		×	High Level Us UX Design Do
		□ Eng Tech Spec Written			Eng Tech Spe
		☐ Eng t-shirt Size & Quarter Commit			Eng t-shirt Siz
		☐ All Sprints Scheduled			All Sprints Sc
		□ All Code Checked In & Tested	Description		
		□ QE Testing Complete			
		<ul> <li>Limited Availability for Flagged Customers</li> </ul>			
		☐ GA for All Customers			





Configuration	Options are defined centrally:		Options are defined per-issue. New issues acquire the Default
	Applicable contexts for scheme:	Edit Configuration  Issue type(s):  Global (all issues)	Applicable contexts for scheme:
		• • • •	Default Value:  Due to what seems like a bug, the Default Value doesn't display  Set Custom Field Defaults  Set the default value for custom field: Milestones  Milestones:  Overview Doc Written  High Level User Stories Writte  UX Design Developed  Eng Tech Spec Written  Eng t-shirt Size & Quarter Cor  All Sprints Scheduled  Set Default Cancel

Customizability	n/a	n/a
Internal	customfieldvalue.stringvalue stores ID references to predefined options:	customfield.textvalue stores one JSON field per issue,
Internal Database Representation		
	customfieldvalue.stringvalue stores ID references to predefined options:  11202 11201	customfield.textvalue stores one JSON field per issue, v {{"id":"Overview Doc Written","type":"done"},{"id":"High Level U- Tech Spec Written","type":"todo"},{"id":"Eng t-shirt Size & Quar
Database	11202 11201	
Database	11202 11201	[("id":"Overview Doc Written","type":"done"),("id":"High Level U Tech Spec Written","type":"todo"),("id":"Eng t-shirt Size & Quar
Database Representation	11202 11201	
Database Representation	11202 11201	[("id":"Overview Doc Written","type":"done"),("id":"High Level U Tech Spec Written","type":"todo"),("id":"Eng t-shirt Size & Quar
Database Representation	11202 11201	[("id":"Overview Doc Written","type":"done"),("id":"High Level U Tech Spec Written","type":"todo"),("id":"Eng t-shirt Size & Quar
Database Representation	11202 11201	[("id":"Overview Doc Written","type":"done"),("id":"High Level U Tech Spec Written","type":"todo"),("id":"Eng t-shirt Size & Quar
Database Representation	11202 11201	[("id":"Overview Doc Written","type":"done"),("id":"High Level U Tech Spec Written","type":"todo"),("id":"Eng t-shirt Size & Quar

I would recommend Checklist for JIRA, being the only plugin allowing centrally managed Task list definitions (i.e. where you can edit a Task definition once, and see the change across all issues), while still allowing per-issue tasks. Checklist for JIRA also comes with a nice set of JQL filters, and the ability to mark tasks 'Mandatory' and other features. Finally, the fact that it uses the standard custom field API is preferable to plugins that become unique snowflakes by implementing their own database tables.

If you don't mind task lists differing per issue, the free ToDo list custom field plugin does a great job.

The Simple Tasklists and Issue Actions Todo plugins implement their own database back-ends, rather than using the custom field API. As a result, neither allows for searching of issues by todo status, or displaying todo status in gadgets. Neither takes the opportunity to make task definitions reusable. Simple Tasklists is interesting, in that it uses the flexibility of a custom back-end to allow for wiki-like interspersing of tasks with description text and comments.

I shall endeavour to keep this page updated – please drop me a line if anything is incorrect or outdated.