

Custom fields

Give your tasks the extra information your team works with - a reference number, a review result, an acceptance criterion - as real fields instead of text inside the description.

Overview

Every **team** defines its own fields for its tasks. A task shows the fields of the team it belongs to, nothing else. The five types are the ones the workflow designer knows, plus a long text:

Type	Holds	Good for
Text	one line, up to 255 characters	reference or ticket number, customer PO
Long text	several lines	"Done when" criteria, notes for the reviewer
Number	a number, decimals allowed	story points, amount, hours
Date	a calendar date	review date, go-live
Dropdown	one of a list of options	review result, severity, release train

Fields can also be marked **required before the task can be finished**.

Projects can have fields too (one list for all projects, defined by an RC Admin), but today only through the REST API and MCP, not in the project pages. Use task fields for everything day-to-day.

1. Define fields

Open **Team Settings** of your team and find the card **Custom fields**. It is there for the team lead, the deputy and RC Admins.

- 1 Click **Add field**.
- 2 Enter a **label** (what people see). The **key** is filled in from it: letters, digits and `_`, starting with a letter. The key is what the API and the filters use.
- 3 Pick the **type**. For a dropdown, write the options one per line - either `value` or `value|Label` (the label is shown, the value is stored).
- 4 Tick **Required** if a task must not be finished while the field is empty.

The arrows move a field up or down - that is the order in the task. **Archive** hides a field everywhere but keeps its values and its key; **Restore** brings it back. Fields are never deleted.

Changing key or type. You can rename the key later, but API clients and filters that use the old key must follow. A new type converts the values already entered (a number 3.5 becomes the text "3.5", a text that reads like a date becomes a date). It is all or nothing: if even one value does not fit, nothing is changed and you are told how many do not fit.

2. Enter values

- **In the task editor** the card **Custom fields** sits under the Task card. Each field is saved when you leave it; a value that does not fit (a date that does not exist) is refused with the reason and the field is marked. Required fields carry a *****.
- **In the Quick Insert Task window** (Shift+Q) the fields appear in their own card as soon as a team is chosen - also for the team that was pre-selected from your last use. Change the team and the fields change with it.
- **In the backlog item form** the fields of the chosen team appear the same way.

Nothing is required when a task is *created*. Required only means: not finished yet.

A task that moves to another team shows that team's fields. The old team's values are hidden, not lost - they are back when the task moves back.

3. Required before done

If a field is marked required and a task of that team has no value in it, the task cannot be finished. Whoever presses **Done** gets a message naming the empty fields. In the editor they are marked and the page scrolls to them; in a list, a click on the message opens the task. The same rule holds for the REST API: `PUT /tasks/{id}` with `done: 1` is refused with a 400 naming the fields - send the values in `custom_fields` in the same request to fill and finish in one call.

A task that is moved to another team in the same request is judged by the team it ends up in.

Not affected: personal ToDo items, and work that nobody finishes by hand - vacations and meetings closed by their modules, invoices, tasks closed by the autoclose job, a project completed as a whole, a Jira history import.

4. Lists and search

- The task lists have a **Custom fields** column. It is **off by default**: tick it in the column chooser (the columns button above the list). It then shows up to three filled fields per task and "+n" for more; the tooltip has the full text. It is not shown in the minimal row view or on narrow screens, and it exists only if some task on the page has a value.
- The **search box** above a list also finds a task by the *values* of its fields (not by the labels).

5. REST API and MCP

Definitions: `GET /custom-fields?team_id=...`, `POST /custom-fields`, `PUT /custom-fields/{id}` (label, key, type, options, required_for_done, position, active), `DELETE /custom-fields/{id}` (archives). Values: tasks and projects carry a `custom_fields` object, in answers and in `POST / PUT` :

```
curl -u USER:TOKEN -X PUT https://YOUR-INSTANCE/api/v1/tasks/123 \
-H 'Content-Type: application/json' \
-d '{"custom_fields": {"review": "ok", "review_on": "2026-11-10"}}'
```

An empty value clears a field. Filtering works on `GET /tasks` :

```
GET /tasks?team_id=7&cf[review]=ok
```

```
GET /tasks?team_id=7&cf[review_on][from]=2026-11-01&cf[review_on][to]=2026-11-30
```

Dropdown and text fields match exactly; numbers and dates take `from` and/or `to` . The hosted MCP endpoint has the same as tools (`list_custom_fields` , `get_custom_field` , `create_custom_field` , `update_custom_field` , `archive_custom_field` , and `custom_fields / cf` arguments on the task and project tools). See the MCP guide.

6. The AI assistant

When the assistant lists your open tasks or the most important open tasks, it sees their filled custom fields (for example *Review: Approved; Review on: 10/11/2026*), so a question like "which of my tasks are waiting for review?" can be answered. It sees only what you can see, and private tasks stay private.

Rules and limits

- Fields are archived, never deleted; an archived field keeps its key, so the key cannot be used again.
- Task fields belong to one team. Fields are not shared between teams and there are no company-wide task fields.
- Custom fields are not part of reports and exports yet.
- Only an RC Admin defines project fields, and only through the API for now.