

Track work item progress in GitLab

Background

Work item tracking tools play a key role in collaborative development environments like The Good Docs Project (TGDP). They enable:

- Team members to keep tabs on the steps they need to carry out next
- Everyone to stay in the loop on how things are going

If you've ever used tracking software like Atlassian's Jira, then you know all about these benefits. But no worries if you haven't. You'll soon experience the upsides of work item tracking as you progress in your work with TGDP.

Overview

This guide will help you track task progress with TGDP. You will use GitLab work items to:

1. [Mark your progress](#) through preset task stages.
 - a. Use **checklists** to keep action steps in focus.
 - b. Coordinate with your team lead to update **phase labels**; this will register your task's status in the larger flow of work at TGDP.
2. [Log key developments](#) by posting comments.
3. [Consult your work item history](#) as needed in the Activity section.
4. [Close your work item](#) upon completion of your work.

Before you start

To get the most from this guide, be sure you've learned how to:

- [Use GitLab to find your first task](#)
- [Get your work rolling using GitLab](#)

1. Mark your progress

Use checklists

Each task you work on at TGDP contains several steps, but you can't work on everything at once! To help make work items manageable, TGDP team leaders often parse out major stages with **checklists**.

Use these checklists to mark when you have completed a preset stage/step. When you select a box:

- You can then focus on completing the next checklist item.

When you reach a new phase in your work on a task—such as progressing from a research to a drafting phase—notify your working group lead. They will advance any **phase label** associated with your issue to the next stage, as appropriate.

GitLab work item labels help TGDP leaders organize, track, and see larger trends in task progress at the project level.

The Good Docs Project / Knowledge base / Issues / #21

GitLab Duo Chat

Example issue, for illustration purposes

Issue created 29 minutes ago by **Thad Fowler**

This is an **Article** for the Knowledge Base that is part of the larger **Epic** project: Onboarding.

The goal of this article is to help new contributors use issues in GitLab for project management purposes. It assumes readers are familiar with how to [use GitLab to find a first task](#) and will guide them to use issues as the source of truth for coordinating work and tracking progress over the lifecycle of a work item.

Link to Google Doc drafts

Our project composes base files in Google Docs that are owned and maintained by the project leads. The Good Docs Project owns these files so that we can maintain our project archive and history.

With that in mind, the project has pre-generated Google Doc files for you to use as you are researching and drafting your project. These files include a starting point for the structure of each file that should help you as you draft the documents:

☐ Draft file - [Example Google Doc Draft](#)

Bookmark this file and use it as you work on your project.

Status

To do

Assignees

Thad Fowler

Labels

kb phase
 To Do
 wg
 [Knowledge base >](#)

Parent

None

Weight

None

Milestone

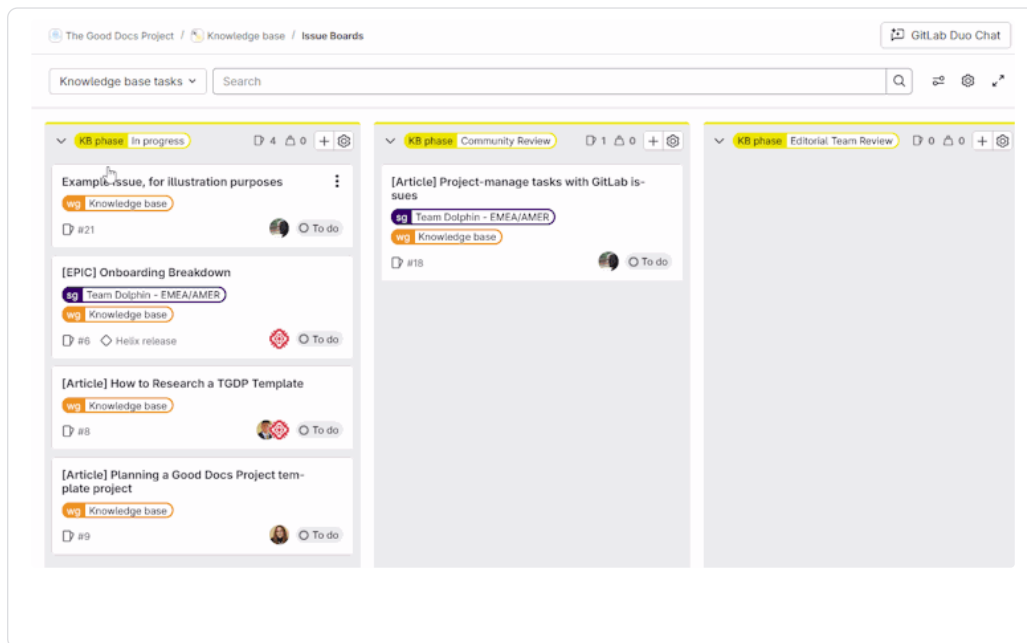
None

Iteration

None

Dates

You can also drag and drop a work item from one phase to another within an issue board. The phase label updates automatically:



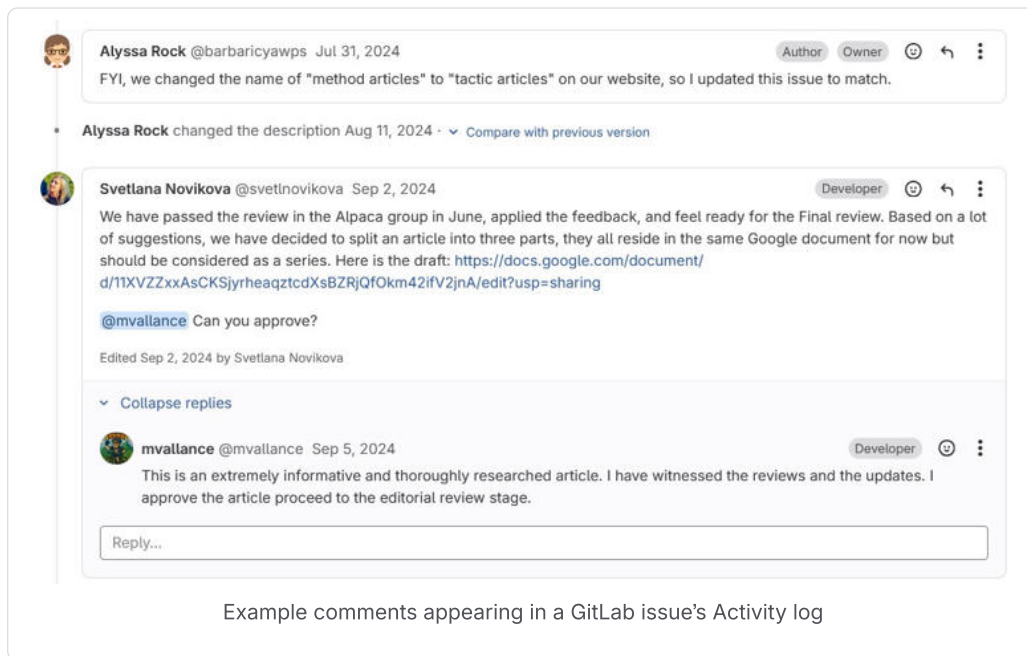
2. Log key developments

As you make headway on your task, you will take various actions that the standard phase labels don't illuminate. Keep everyone on the same page about these developments using your work item's **Comments** functionality.

What information should you record? As a rule of thumb, post any information a person would require if they needed to jump in, clearly grasp the actions you've taken so far, and pick up where you left off. In so doing, you enhance [transparency](#) and improve collaboration:

1. Scroll to the bottom of the work item (beneath the issue's Activity log).
2. Type your comment in the text input field. You can optionally include attachments or imagery.
3. Click the **Comment** button to post.
4. View your posted comment in the Activity log. If needed, you can edit your comment.

Check out an example set of work item comments:



Notice that you can use comments to **directly communicate** with TGDP community members:

1. Use the **@username** format to mention one or more community members in a comment.
2. GitLab notifies the user of your post with a to-do item and email.
3. Subsequent responses to the original comment display in a thread in the work item's Activity log.

While the Good Docs community uses Discord for much of its messaging, you should use GitLab to communicate about need-to-know task-related information—especially when people will benefit from referring to that communication later on.

3. Consult your work item history

With TGDP's bi-yearly release cycles, you may work on your task over a [longish stretch of time](#). During that span you may, on occasion, find it helpful to jog your memory about important details that arose earlier in the process but which you lost track of.

At such moments, you can refer to a record of prior developments in the **Activity section** presented at the bottom of your GitLab issue. All data updates associated with your work item are automatically logged in that location.

4. Close your issue

After progressing through all phases of your task, finalizing all deliverables, and meeting all conditions noted in the definition of done, you are ready to close your work item. Confer with your working group lead to confirm that completed status, and then one of you can close the work item in GitLab:

1. Select the **Work item actions** dropdown in the top-right of the issue (marked with three vertical dots).
2. Select **Close work item**.

See also

- GitLab Docs: [Work items](#)