Best practices for Projects

In this article

Break down large issues into smaller issues

Communicate

Make use of the description and README

Use views

Have a single source of truth

Use automation

Use different field types

Learn tips for managing your projects.

You can use Projects to manage your work on GitHub, where your issues and pull requests live. Read on for tips to manage your projects efficiently and effectively. For more information about Projects, see "About Projects."

Break down large issues into smaller issues 🔗

Breaking a large issue into smaller issues makes the work more manageable and enables team members to work in parallel. It also leads to smaller pull requests, which are easier to review.

To track how smaller issues fit into the larger goal, use task lists, milestones, or labels. For more information, see "About task lists", "About milestones", and "Managing labels."

Communicate 🔗

Issues and pull requests include built-in features to let you easily communicate with your collaborators. Use @mentions to alert a person or entire team about a comment. Assign collaborators to issues to communicate responsibility. Link to related issues or pull requests to communicate how they are connected.

Make use of the description and README 🔗

Use your project's description and README to share information about the project.

For example:

- Explaining the purpose of the project.
- Describing the project views and how to use them.
- Including relevant links and people to contact for more information.

Project READMEs support Markdown which allows you to use images and advanced formatting such as links, lists, and headers.

For more information, see "Creating a project."

Use views ⚯

Use project views to look at your project from different angles.

For example:

- Filter by status to view all un-started items
- Group by a custom priority field to monitor the volume of high priority items
- Sort by a custom date field to view the items with the earliest target ship date

For more information, see "Changing the layout of a view."

Have a single source of truth ⚯

To prevent information from getting out of sync, maintain a single source of truth. For example, track a target ship date in a single location instead of spread across multiple fields. Then, if the target ship date shifts, you only need to update the date in one location.

Projects automatically stay up to date with GitHub data, such as assignees, milestones, and labels. When one of these fields changes in an issue or pull request, the change is automatically reflected in your project.

Use automation ⚯

You can automate tasks to spend less time on busy work and more time on the project itself. The less you need to remember to do manually, the more likely your project will stay up to date.

Projects offers built-in workflows. For example, when an issue is closed, you can automatically set the status to "Done." You can also configure built-in workflows to automatically archive items when they meet certain criteria and to automatically add items from a repository when they match a filter.

Additionally, GitHub Actions and the GraphQL API enable you to automate routine project management tasks. For example, to keep track of pull requests awaiting review, you can create a workflow that adds a pull request to a project and sets the status to "needs review"; this process can be automatically triggered when a pull request is marked as "ready for review."

- For more information about the built-in workflows, see "Using the built-in automations."
- For more information about automatically archiving items, see "Archiving items automatically."
- For more information about automatically adding items, see "Adding items automatically."
- For an example workflow, see "Automating Projects using Actions."
- For more information about the API, see "Using the API to manage Projects."
- For more information about GitHub Actions, see "GitHub Actions documentation."

Use different field types ⚯

Take advantage of the various field types to meet your needs.

Use an iteration field to schedule work or create a timeline. You can group by iteration to see if items are balanced between iterations, or you can filter to focus on a single iteration. Iteration fields also let you view work that you completed in past iterations,

which can help with velocity planning and reflecting on your team's accomplishments. Iteration fields also support breaks to show when you and your team are taking time away from their iterations. For more information, see "About iteration fields."

Use a single select field to track information about a task based on a preset list of values. For example, track priority or project phase. Since the values are selected from a preset list, you can easily group or filter to focus on items with a specific value.

For more information about the different field types, see "Understanding fields."