About Projects

Projects is an adaptable, flexible tool for planning and tracking work on GitHub.

About Projects 🔗

A project is an adaptable spreadsheet, task-board, and road map that integrates with your issues and pull requests on GitHub to help you plan and track your work effectively. You can create and customize multiple views by filtering, sorting, grouping your issues and pull requests, visualize work with configurable charts, and add custom fields to track metadata specific to your team. Rather than enforcing a specific methodology, a project provides flexible features you can customize to your team's needs and processes.

Staying up-to-date 🔗

Your projects are built from the issues and pull requests you add, creating direct references between your project and your work. Information is synced automatically to your project as you make changes, updating your views and charts. This integration works both ways, so that when you change information about a pull request or issue in your project, the pull request or issue reflects that information. For example, change an assignee in your project and that change is shown in your issue. You can take this integration even further, group your project by assignee, and make changes to issue assignment by dragging issues into the different groups.

Adding metadata to your items 🔗

You can use custom fields to add metadata to your issues, pull requests, and draft issues and build a richer view of item attributes. You're not limited to the built-in metadata (assignee, milestone, labels, etc.) that currently exists for issues and pull requests. For example, you can add the following metadata as custom fields:

- A date field to track target ship dates.
- A number field to track the complexity of a task.
- A single select field to track whether a task is Low, Medium, or High priority.
- A text field to add a quick note.
- An iteration field to plan work week-by-week, including support for breaks.

Automating your projects 🔗

There are a number of ways you can add automation to your project. Built-in workflows allow you to automatically set fields when items are added or changed, and you can also configure your project to automatically archive items when they meet certain criteria and automatically add items from a repository when they match set criteria. For more information, see "Using the built-in automations."

You can also use the GraphQL API and GitHub Actions to take even greater control of your project. For more information, see "Using the API to manage Projects" and "Automating Projects using Actions."

Exploring the relationships between issues 🔗

> **Note:** Tasklists are in private beta and subject to change. We have currently paused onboarding to the beta.

You can use tasklists to build hierarchies of issues, dividing your issues into smaller subtasks, and creating new relationships between your issues. For more information, see "[Creating a tasklist](#)."

These relationships are displayed on the issue, as well as the Tracked by and Tracks fields in your projects. You can filter by issues which are tracked by another issue, and you can also group your table views by the Tracked by field to show all parent issues with a list of their subtasks.

Viewing your project from different perspectives 🔗

Quickly answer your most pressing questions by tailoring your project view to give you the information you need. You can save these views, allowing you to quickly return to them as needed and make them available to your team. Views not only let you scope down the items listed but also offer two different layout options.

You can view your project as a high-density table layout, as a kanban board, or a timeline-style roadmap. For more information about the different layout options, see "[Changing the layout of a view](#)."