

# Best practices for Projects

Learn tips for managing your projects.

## In this article

Communicate across your issues and pull requests

### **Break down large issues into smaller issues**

Make use of the description, README, and status updates to share information about a project

Create customized views of your project items

Use different field types to add metadata to your project items

Use automation to keep your projects up to date automatically

Create charts and insights to visualize and share progress

Create project templates to standardize your workflows

Link projects to teams and repositories

Have a single source of truth

Further reading

---

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](#).

## Communicate across your issues and pull requests

---

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.

# Break down large issues into smaller issues

---

Breaking a large issue into smaller issues makes the work more manageable and enables team



≡ GitHub Issues / Projects / Learning about Projects / Best practices for Projects

between your issues. You can create multiple levels of sub-issues that accurately represent your project by breaking down tasks into exactly the amount of detail that you and your team require. See [Adding sub-issues](#) and [Browsing sub-issues](#).

You can also use issue types to classify work in repositories across the organization alongside sub-issues. For more information, see [Managing issue types in an organization](#).

To ensure efficient progress, clearly define which issues are blocked by, or blocking, other issues. See [Creating issue dependencies](#).

To track how smaller issues fit into the larger goal, use milestones or labels. For more information, see [About milestones](#) and [Managing labels](#).

## Make use of the description, README, and status updates to share information about a project

---

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](#).

You can also share high-level updates with other users of your project by posting status updates. Status updates allow you to mark the project with a status, such as "On track" or "At risk", set start and target dates, and share written updates with your team. For more information, see [Sharing project updates](#).

# Create customized views of your project items

Use project views to look at your project from different angles using the table, board, and roadmap layout. Views allow you to manage your team backlog, weekly iterations, team roadmaps, and plans for a feature release, just to name a few.

For example, you can customize views by:

- Filtering by status to view all un-started items
- Grouping by a custom priority field to monitor the volume of high priority items
- Sorting by a custom date field to view the items with the earliest target ship date
- Slicing by assignee to view team capacity
- Showing a field sum for an estimate to highlight complexity for a group of items
- Adding a column limit to a board column to maintain focus

Here is an example table layout:

🔒 **OctoArcade Invaders**

On track

Team Backlog

Daily stand-up board

Roadmap

Team Epics

By priority

Bug triage

My work

All

Team Capacity

Done

Milestones

has:sub-issues-progress is:open

16

Discard

Save

| Type                  | Title   | Sub-issues progress | Status      | Assignees       | Linked pull requests | Priority |
|-----------------------|---|---------------------|-------------|-----------------|----------------------|----------|
| Squad 1 3 Estimate: 4 |   |                     |             |                 |                      |          |
| 1 Task                | Updates to velocity of the ship and alien mo... #802    | 1 / 2 50%           | Planning    | azenMatt        | #374                 | Urgent   |
| 2 Epic                | Task Management #39                                     | 2 / 6 33%           | Review      | rileybroughten  |                      | High     |
| 3 Bug                 | Search functionality returns no results for ex... #1411 | 0 / 1 0%            | Triage      | rileybroughten  | #1218                | Low      |
| + Add item            |   |                     |             |                 |                      |          |
| Squad 2 4 Estimate: 3 |   |                     |             |                 |                      |          |
| 4 Task                | Start explorations for game site #801                   | 1 / 2 50%           | Building    | ajashams and... | #993                 | High     |
| 5 Initiative          | Social Media Campaign #19                               | 0 / 4 0%            | Not Started | LisaKr          |                      | High     |
| 6 Epic                | Integration and Automation #42                          | 0 / 5 0%            | Not Started | azenMatt        |                      | Low      |
| 7 Epic                | Reporting and Analytics #41                             | 0 / 8 0%            | Planning    | sunnyi101       |                      | Medium   |
| + Add item            |   |                     |             |                 |                      |          |
| Squad 3 4 Estimate: 2 |   |                     |             |                 |                      |          |

Here is an example board layout:

**OctoArcade Invaders** On track

Team Backlog | **Daily stand-up board** | Roadmap | Team Epics | By priority | Bug triage | My work | All | Team Capacity | Done | Milestones

Search: cycle:@current 25 Discard Save

- Triage** (2 items, Estimate: 3)
  - planning-tracking-demo #802: Updates to velocity of the ship and alien movements (Squad 1, #374, Task)
  - productivity #46: Copilot assignee icon is cut off on the left (#68)
- Not Started** (6/5 items, Estimate: 9)
  - planning-tracking-demo #797: Easter egg unlocking new paid levels (Squad 3, Initiative)
  - planning-tracking-demo #829: Hero site - Copy (Squad 3, Bug)
  - planning-tracking-demo #1394: Update testing framework (#1218, Bug)
  - planning-tracking-demo #1511: Build a daily leaderboard (Feature)
- Planning** (7 items, Estimate: 10)
  - planning-tracking-demo #1315: Investigate error message (Squad 2, Bug)
  - planning-tracking-demo #828: Acquire domain for launch (Squad 3, Bug)
  - planning-tracking-demo #1376: Notifications stopped working (Bug)
  - planning-tracking-demo #1304: Make sure that it delights customers (Squad 3, Task)
- Building** (5/3 items, Estimate: 110)
  - planning-tracking-demo #808: General bug fixes from Alpha feedback (Squad 3, #992, Batch)
  - planning-tracking-demo #820: Create user account and subscription (Squad 1, Initiative)
  - planning-tracking-demo #832: Final creative shots from game (Squad 3, Bug)
  - planning-tracking-demo #1323: Finalize upsell model (Squad 1, Batch)

Here is an example roadmap layout:

**OctoArcade Invaders** On track

Team Backlog | Daily stand-up board | **Roadmap** | Team Epics | By priority | Bug triage | My work | All | Team Capacity | Done | Milestones

Search: has:cycle 85 Discard Save

June 2025 | July 2025 | August 2025

Markers | Cycle | Date fields | Quarter | Today

**Squad 1** (6 items)
 

- Write PR based on content plan #14
- Updates to velocity of the ship and alien movements #802
- Create user account and subscription #820
- Finalize upsell model #1323
- Task Management #39
- Search functionality returns no results for existing products #1411

**Squad 2** (9 items)
 

- Investigate error message #1315
- Final UX tweaks #1312

**GA release** (Milestone)

For more information, see [Customizing views in your project](#).

## Use different field types to add metadata to your project items

Take advantage of the various field types to meet your needs and add metadata to your issues, pull requests, and draft issues for richer views. 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

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 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. 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](#).

## Use automation to keep your projects up to date automatically

---

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](#).

## Create charts and insights to visualize and share progress

---

You can use insights for Projects to view, create, and customize charts that use the items added to your project as their source data. You can apply filters to the default chart and also create your own charts. When you create a chart, you set the filters, chart type, the information displayed, and the chart is available to anyone that can view the project.

For more information, see [About insights for Projects](#).

## Create project templates to standardize your workflows

---

You can create project templates for your organization, or set a project as a template, to share a pre-configured project with other people in your organization which they can then use as the base for their projects. Project templates include the views, custom fields, draft issues and associated fields, configured workflows (except any auto-add workflows), and insights.

For more information, see [Managing project templates in your organization](#).

## Link projects to teams and repositories

---

You can add projects to your team to give the whole team collaborator access to their projects. When you add a project to a team, that project is listed on the team's projects page, making it easier for members to identify which projects a particular team uses.

For more information, see [Adding your project to a team](#).

You can also add projects to a repository that is owned by the same user or organization that owns the project.

For more information, see [Adding your project to a repository](#).

## 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.

## Further reading

---

- [About Projects](#)
- [Quickstart for Projects](#)

### Legal

© 2025 GitHub, Inc. [Terms](#) [Privacy](#) [Status](#) [Pricing](#) [Expert services](#) [Blog](#)