

Planning and tracking work for your team or project

In this article

Introduction

Creating a repository

Communicating repository information

Creating issue templates

Opening issues and using task lists to track work

Making decisions as a team

Using labels to highlight project goals and status

Adding issues to a project board

Next steps

The essentials for using GitHub's planning and tracking tools to manage work on a team or project.

Introduction

You can use GitHub repositories, issues, project boards, and other tools to plan and track your work, whether working on an individual project or cross-functional team.

In this guide, you will learn how to create and set up a repository for collaborating with a group of people, create issue templates and forms, open issues and use task lists to break down work, and establish a project board for organizing and tracking issues.

Creating a repository

When starting a new project, initiative, or feature, the first step is to create a repository. Repositories contain all of your project's files and give you a place to collaborate with others and manage your work. For more information, see "[Creating a new repository](#)."

You can set up repositories for different purposes based on your needs. The following are some common use cases:

- **Product repositories:** Larger organizations that track their work and goals around specific products may have one or more repositories containing the code and other files. These repositories can also be used for documentation, reporting on product health or future plans for the product.
- **Project repositories:** You can create a repository for an individual project you are working on, or for a project you are collaborating on with others. For an organization that tracks work for short-lived initiatives or projects, such as a consulting firm, there is a need to report on the health of a project and move people between different projects based on skills and needs. Code for the project is often contained in a single repository.
- **Team repositories:** For an organization that groups people into teams, and brings projects to them, such as a dev tools team, code may be scattered across many

repositories for the different work they need to track. In this case it may be helpful to have a team-specific repository as one place to track all the work the team is involved in.

- **Personal repositories:** You can create a personal repository to track all your work in one place, plan future tasks, or even add notes or information you want to save. You can also add collaborators if you want to share this information with others.

You can create multiple, separate repositories if you want different access permissions for the source code and for tracking issues and discussions. For more information, see "[Creating an issues-only repository](#)."

For the following examples in this guide, we will be using an example repository called Project Octocat.

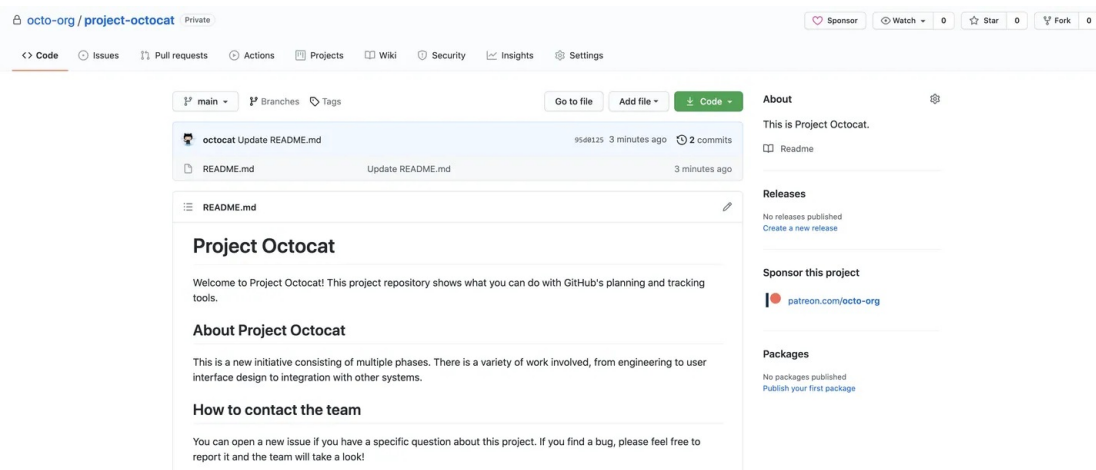
Communicating repository information [↗](#)

You can create a README.md file for your repository to introduce your team or project and communicate important information about it. A README is often the first item a visitor to your repository will see, so you can also provide information on how users or contributors can get started with the project and how to contact the team. For more information, see "[About READMEs](#)."

You can also create a CONTRIBUTING.md file specifically to contain guidelines on how users or contributors can contribute and interact with the team or project, such as how to open a bug fix issue or request an improvement. For more information, see "[Setting guidelines for repository contributors](#)."

README example [↗](#)

We can create a README.md to introduce our new project, Project Octocat.



Creating issue templates [↗](#)

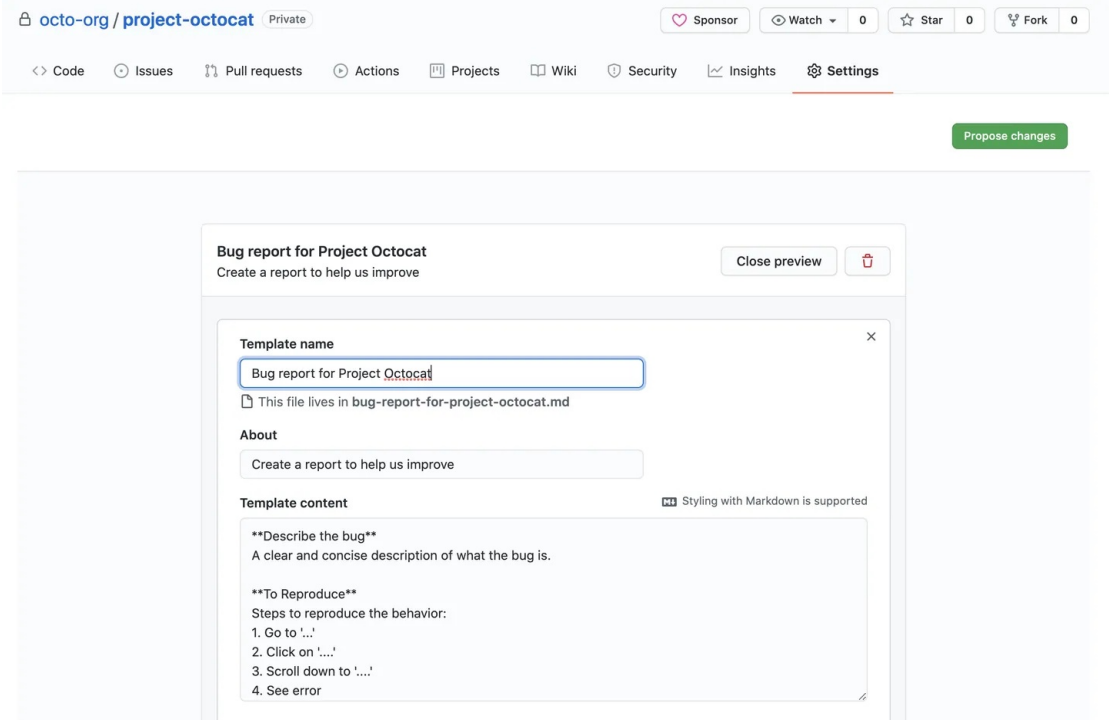
You can use issues to track the different types of work that your cross-functional team or project covers, as well as gather information from those outside of your project. The following are a few common use cases for issues.

- **Release tracking:** You can use an issue to track the progress for a release or the steps to complete the day of a launch.
- **Large initiatives:** You can use an issue to track progress on a large initiative or project, which is then linked to the smaller issues.
- **Feature requests:** Your team or users can create issues to request an improvement to your product or project.
- **Bugs:** Your team or users can create issues to report a bug.

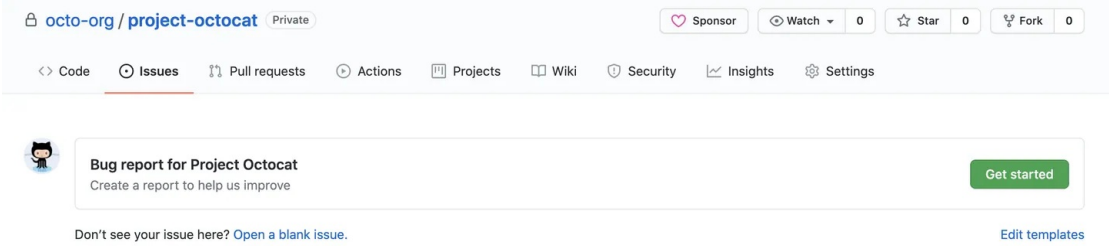
Depending on the type of repository and project you are working on, you may prioritize certain types of issues over others. Once you have identified the most common issue types for your team, you can create issue templates and forms for your repository. Issue templates and forms allow you to create a standardized list of templates that a contributor can choose from when they open an issue in your repository. For more information, see "[Configuring issue templates for your repository](#)."

Issue template example [↗](#)

Below we are creating an issue template for reporting a bug in Project Octocat.



Now that we created the bug report issue template, you are able to select it when creating a new issue in Project Octocat.



Opening issues and using task lists to track work [↗](#)

You can organize and track your work by creating issues. For more information, see "[Creating an issue](#)."

Issue example [↗](#)

Here is an example of an issue created for a large initiative, front-end work, in Project Octocat.

[Code](#) [Issues 1](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#) [Settings](#)

Front-end work for Project Octocat #1

EditNew issue

Open octocat opened this issue 1 minute ago · 0 comments

octocat commented 1 minute ago

Description

We need to complete multiple tasks to make sure the front-end works smoothly and as designed.

Tasks

1. Design user interface

2. Implement personalized user interactions

3. Implement for mobile

4. Integrate with back-end

Assignees

No one—assign yourself

Labels

None yet

Projects

None yet

Milestone

No milestone

Linked pull requests

Successfully merging a pull request may close this issue.

None yet

Notifications

Unsubscribe

You're receiving notifications because you authored the thread

WritePreview

H B I

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

Close issue

Comment

Task list example [↗](#)

You can use task lists to break larger issues down into smaller tasks and to track issues as part of a larger goal. Task lists have additional functionality when added to the body of an issue. You can see the number of tasks completed out of the total at the top of the issue, and if someone closes an issue linked in the task list, the checkbox will automatically be marked as complete. For more information, see "[About task lists](#)."

Below we have added a task list to our Project Octocat issue, breaking it down into smaller issues.

Front-end work for Project Octocat #1

EditNew issue

Open octocat opened this issue 25 minutes ago · 0 comments

octocat commented 25 minutes ago · edited

Description

We need to complete multiple tasks to make sure the front-end works smoothly and as designed.

Tasks

☒ [Project Octocat] Design user interface

☐ [Project Octocat] Develop functionality for personalized user interactions and animations

☐ [Project Octocat] Optimize for mobile

☐ PLACEHOLDER - Integrate with back-end

Assignees

No one—assign yourself

Labels

None yet

Projects

None yet

Milestone

No milestone

Linked pull requests

Successfully merging a pull request may close this issue.

None yet

Notifications

Unsubscribe

You're receiving notifications because you authored the thread

Making decisions as a team [↗](#)

You can use issues and discussions to communicate and make decisions as a team on planned improvements or priorities for your project. Issues are useful when you create them for discussion of specific details, such as bug or performance reports, planning for the next quarter, or design for a new initiative. Discussions are useful for open-ended brainstorming or feedback, outside the codebase and across repositories. For more information, see "[Communicating on GitHub](#)."

As a team, you can also communicate updates on day-to-day tasks within issues so that everyone knows the status of work. For example, you can create an issue for a large feature that multiple people are working on, and each team member can add updates with their status or open questions in that issue.

Issue example with project collaborators [↗](#)

Here is an example of project collaborators giving a status update on their work on the

Front-end work for Project Octocat #1

Open

octocat opened this issue 17 hours ago · 2 comments

Edit

New issue

octocat commented 17 hours ago · edited

Description

We need to complete multiple tasks to make sure the front-end works smoothly and as designed.

Tasks

☒ [Project Octocat] Design user interface

☐ [Project Octocat] Develop functionality for personalized user interactions and animations

☐ [Project Octocat] Optimize for mobile

☐ PLACEHOLDER - Integrate with back-end

saritai added the front-end label 17 hours ago

codercat commented 17 hours ago

Author

I just looked at the completed user interface design and it looks great! My status is that I am still working on our user interaction feature and will probably be focused on that for the next few days.

octocat commented 17 hours ago

Author

I have been breaking down all the steps it will take to integrate with the back-end and will create multiple issues for that work tomorrow!

Assignees

octocat

Labels

front-end

Projects

None yet

Milestone

No milestone

Linked pull requests

Successfully merging a pull request may close this issue.

None yet

Notifications

Customize

Unsubscribe

You're receiving notifications because you authored the thread.

2 participants

Using labels to highlight project goals and status

You can create labels for a repository to categorize issues, pull requests, and discussions. GitHub also provides default labels for every new repository that you can edit or delete. Labels are useful for keeping track of project goals, bugs, types of work, and the status of an issue.

For more information, see "[Managing labels](#)."

Once you have created a label in a repository, you can apply it on any issue, pull request or discussion in the repository. You can then filter issues and pull requests by label to find all associated work. For example, find all the front end bugs in your project by filtering for issues with the front-end and bug labels. For more information, see "[Filtering and searching issues and pull requests](#)."

Label example

Below is an example of a front-end label that we created and added to the issue.

octocat commented 30 minutes ago · edited

Description

We need to complete multiple tasks to make sure the front-end works smoothly and as designed.

Tasks

☒ [Project Octocat] Design user interface

☐ [Project Octocat] Develop functionality for personalized user interactions and animations

☐ [Project Octocat] Optimize for mobile

☐ PLACEHOLDER - Integrate with back-end

octocat added the front-end label 2 minutes ago

Assignees

No one—assign yourself

Labels

front-end

Projects

None yet

Milestone

No milestone

Linked pull requests

Adding issues to a project board

You can use projects on GitHub to plan and track the work for your team. A project is a customizable spreadsheet that integrates with your issues and pull requests on GitHub, automatically staying up-to-date with the information on GitHub. You can customize the layout by filtering, sorting, and grouping your issues and PRs. To get started with

projects, see "[Quickstart for Projects](#)."

Project example

Here is the table layout of an example project, populated with the Project Octocat issues we have created.

Project Octocat

View 1

View 2

New view

Alpha

Give feedback

	Title	Assignees	Status	Labels	Notes	
1	Front-end work for Project Octocat	octocat	In Progress	front-end	Still breaking down the work	
2	[Project Octocat] Optimize for mobile		Todo	front-end		
3	[Project Octocat] Develop functionality for personalized user interactions and animations	codercat	In Progress	front-end		
4	[Project Octocat] Design user interface		Done	front-end		

You can use `Ctrl + Space` to add an item

We can also view the same project as a board.

Project Octocat

View 1

View 2

New view

Alpha

Give feedback

Filter by keyword or by field

No Status

Todo

project-octocat #4

[Project Octocat] Optimize for mobile

In Progress

project-octocat #1

Front-end work for Project Octocat

project-octocat #3

[Project Octocat] Develop functionality for personalized user interactions and animations

Done

project-octocat #2

[Project Octocat] Design user interface

You can also use the existing projects (classic) on GitHub to plan and track your or your team's work. Project boards are made up of issues, pull requests, and notes that are categorized as cards in columns of your choosing. You can create project boards for feature work, high-level roadmaps, or even release checklists. For more information, see "[About projects \(classic\)](#)."

Project board example

Below is a project board for our example Project Octocat with the issue we created, and the smaller issues we broke it down into, added to it.

octo-org / project-octocat

Sponsor

Watch

Star

Fork

Code

Issues

Pull requests

Actions

Projects

Wiki

Security

Insights

Settings

Project Octocat Board

Updated 2 minutes ago

Filter cards

+ Add cards

Fullscreen

Menu

To do

[Project Octocat] Optimize for mobile

#4 opened by octocat

In progress

1 of 4

Front-end work for Project Octocat

#1 opened by octocat

front-end

[Project Octocat] Develop functionality for personalized user interactions and animations

#3 opened by octocat

Done

[Project Octocat] Design user interface

#2 opened by octocat

+ Add column

Next steps

You have now learned about the tools GitHub offers for planning and tracking your work, and made a start in setting up your cross-functional team or project repository! Here are some helpful resources for further customizing your repository and organizing your work.

- "[About repositories](#)" for learning more about creating repositories
- "[Tracking your work with issues](#)" for learning more about different ways to create and manage issues
- "[About issue and pull request templates](#)" for learning more about issue templates
- "[Managing labels](#)" for learning how to create, edit and delete labels
- "[About task lists](#)" for learning more about task lists
- "[About Projects](#)" for learning more about projects
- "[Changing the layout of a view](#)" for learning how to customize views for projects

- "[About projects \(classic\)](#)" for learning how to manage project boards

Legal

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Status](#) [Pricing](#) [Expert services](#) [Blog](#)