



Planning and tracking work with GitHub

Childhood Cancer
Data Lab

x





Objectives

- Describe some of the general benefits of an issue tracking system
- Review the features of GitHub issues, including issue templates
- Discuss what makes a “good” issue
- Illustrate how issue scope can have broader implications
- Briefly mention how we currently use GitHub issues to plan our work at the Data Lab

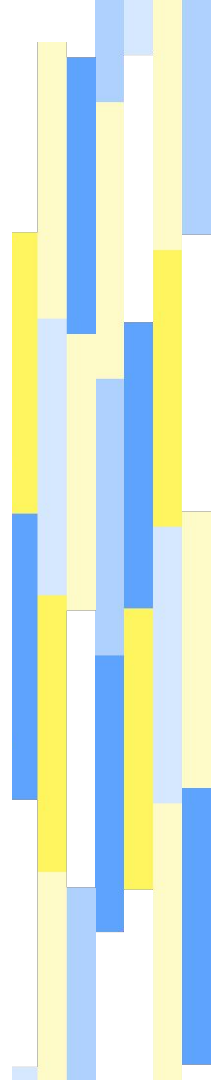


How issue tracking helps

- Reduces the cognitive burden associated with tracking tasks in our head
- Allows us to have efficient conversations about how to tackle work
- Gives us the ability to surface known limitations that we don't intend to fix
- For scientific projects that are public or will be public eventually, there's the additional benefit of capturing scholarship that sometimes remains hidden



GitHub issues allow us
to track issues
alongside our code



Tracking and planning work with GitHub #9

Edit New issue

Open jaclyn-taroni opened this issue last week · 1 comment



jaclyn-taroni commented last week

This issue tracks a lesson on tracking and planning work with GitHub. It will introduce GitHub issues and cover the following:

- What makes a "good" issue?
 - Small in scope
 - Contains enough context that someone else could conceivably carry out the work
 - It will eventually get assigned to someone
- Why are small issues good issues
 - Helps spread work across a team
 - May help avoid merge conflicts
 - Leads to pull requests that are more manageable to review
 - Small issues lead to small wins
- Using issue templates to help you write "good issues"
- A brief mention of sprint planning
 - ZenHub and GitHub Projects as tools that exist to help



Assignees

jaclyn-taroni

Labels

slides

Projects

None yet

Milestone

No milestone

Development

Create a branch for this issue or link a pull request.

Notifications

Sign in now to use Zenhub

Unsubscribe

Tracking and planning work with GitHub

Open jaclyn-taroni opened this issue last week · 1 comment

Allow us to write in GitHub-flavored Markdown

Edit New Issue



jaclyn-taroni commented last week

This issue tracks a lesson on tracking and planning work with GitHub. It will introduce GitHub issues and cover the following:

- What makes a "good" issue?
 - Small in scope
 - Contains enough context that someone else could conceivably carry out the work
 - It will eventually get assigned to someone
- Why are small issues good issues
 - Helps spread work across a team
 - May help avoid merge conflicts
 - Leads to pull requests that are more manageable to review
 - Small issues lead to small wins
- Using issue templates to help you write "good issues"
- A brief mention of sprint planning
 - ZenHub and GitHub Projects as tools that exist to help



Assignees

jaclyn-taroni

Labels

slides

Projects

None yet

Milestone

No milestone

Development

Create a branch for this issue or link a pull request.

Notifications

Sign in now to use Zenhub

Unsubscribe

Tracking and planning work with GitHub #9

Edit New Issue

Open jaclyn-taroni opened this issue last week · 1 comment



jaclyn-taroni commented last week

Allow us to assign specific people to carry out a task

This issue tracks a lesson on tracking following:

- What makes a "good" issue?
 - Small in scope
 - Contains enough context that someone else could conceivably carry out the work
 - It will eventually get assigned to someone
- Why are small issues good issues
 - Helps spread work across a team
 - May help avoid merge conflicts
 - Leads to pull requests that are more manageable to review
 - Small issues lead to small wins
- Using issue templates to help you write "good issues"
- A brief mention of sprint planning
 - ZenHub and GitHub Projects as tools that exist to help

Assignees

jaclyn-taroni

Labels

slides

Projects

None yet

Milestone

No milestone

Development

Create a branch for this issue or link a pull request.

Notifications Sign in now to use Zenhub

Unsubscribe

Tracking and planning work with GitHub #9

Edit New Issue

Open jaclyn-taroni opened this issue last week · 1 comment



jaclyn-taroni commented last week

This issue tracks a lesson on tracking and planning work with GitHub. It will introduce GitHub issues and cover the following:

- What makes a "good" issue?
 - Small in scope
 - Contains enough context to understand the problem
 - It will eventually get assigned to someone
- Why are small issues good issues?
 - Helps spread work across team
 - May help avoid merge conflicts
 - Leads to pull requests that are more manageable to review
 - Small issues lead to small wins
- Using issue templates to help you write "good issues"
- A brief mention of sprint planning
 - ZenHub and GitHub Projects as tools that exist to help

Allow us to organize issues with labels, milestones, and projects

Assignees

jaclyn-taroni

Labels

slides

Projects

None yet

Milestone

No milestone

Development

Create a branch for this issue or link a pull request.

Notifications Sign in now to use Zenhub

Unsubscribe

Tracking and planning work with GitHub #9
jaclyn-taroni opened this issue last week · 4 comments

Write Preview H B I ≡ <> 🔗 ⋮ ⋮ ⋮ @ 🗨️ ↶ 🗑️

Leave a comment

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

Close issue Comment

- ✓ **Close as completed**
Done, closed, fixed, resolved
- Close as not planned**
Won't fix, can't repro, duplicate, stale

Remember, contributions to this repository should follow its [code of conduct](#)

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Docs](#) [Contact GitHub](#) [Pricing](#) [API](#) [Training](#) [Blog](#) [Ab](#) [Sign in now to use Zenhub](#)

Allow us to close issues as completed or as not planned

Filters is:issue is:open Labels 11 Milestones 0 [New issue](#)

<input type="checkbox"/>	17 Open ✓ 4 Closed	Author	Label	Projects	Milestones	Assignee	Sort
<input type="checkbox"/>	Final version of schedule #20 opened last week by jaclyn-taroni						
<input type="checkbox"/>	Initial review through getting code merged live-demo slides #17 opened last week by jaclyn-taroni						
<input type="checkbox"/>	How to leave pull request reviews (live demo) live-demo #16 opened last week by jaclyn-taroni						
<input type="checkbox"/>	On being a pull request reviewer (slides) slides #15 opened last week by jaclyn-taroni						
<input type="checkbox"/>	Strategies for more reasonable diffs (live demo with some slides) live-demo slides #14 opened last week by jaclyn-taroni						2
<input type="checkbox"/>	On being a pull request author (slides) slides #13 opened last week by jaclyn-taroni						
<input type="checkbox"/>	Facilitating review at a "systems level" slides						

[Sign in now](#) to use Zenhub

Filters is:open is:issue assignee:jaclyn-taroni Labels 11 Milestones 0 New issue

Clear current search query, filters, and sorts

<input type="checkbox"/>	8 Open	0 Closed	Author	Label	Projects	Milestones	Assignee	Sort
<input type="checkbox"/>	On being a pull request reviewer (slides)	slides						
	#15 opened last week by jaclyn-taroni							
<input type="checkbox"/>	On being a pull request author (slides)	slides						
	#13 opened last week by jaclyn-taroni							
<input type="checkbox"/>	Facilitating review at a "systems level"	slides						
	#12 opened last week by jaclyn-taroni							
<input type="checkbox"/>	Pull request responsibilities shared between authors and reviewers	slides						2
	#11 opened last week by jaclyn-taroni							
<input type="checkbox"/>	Analytical code review overview	slides						2
	#10 opened last week by jaclyn-taroni							
<input type="checkbox"/>	Tracking and planning work with GitHub	slides						
	#9 opened last week by jaclyn-taroni							

Sign in now to use Zenhub

Filters is:open is:issue label:live-demo Labels 11 Milestones 0 New issue

Clear current search query, filters, and sorts

<input type="checkbox"/>	5 Open ✓ 0 Closed	Author	Label	Projects	Milestones	Assignee	Sort
<input type="checkbox"/>	Initial review through getting code merged live-demo slides #17 opened last week by jaclyn-taroni						
<input type="checkbox"/>	How to leave pull request reviews (live demo) live-demo #16 opened last week by jaclyn-taroni						
<input type="checkbox"/>	Strategies for more reasonable diffs (live demo with some slides) live-demo slides #14 opened last week by jaclyn-taroni						2
<input type="checkbox"/>	Working with branches in Git live-demo slides #5 opened last week by jaclyn-taroni						
<input type="checkbox"/>	Git basics live-demo slides #4 opened last week by jaclyn-taroni						1

ProTip! Add no:assignee to see everything that's not assigned.

Sign in now to use Zenhub

sjspielman requested your review on this pull request. Add your review

Performing code review instructor notes #30

Edit <> Code

Open sjspielman wants to merge 2 commits into main from sjspielman/pr-review-instructor-notes

Conversation 0 Commits 2 Checks 0 Files changed 1 +27 -0



sjspielman commented 14 minutes ago

This PR adds instructor notes towards #16, which entails leaving PR reviews as a live demo; this PR might actually be enough to close that issue, since we have AlexsLemonade/2023-chop-training-demo#5 tracking next steps.

I drew on the ideas sketched out in AlexsLemonade/2023-chop-training-demo#5 to create these instructor notes, divvying up types of comments in different places one can leave comments.

Questions for reviewers:

- Are there other pull request views or features that I should mention here?
- I don't expect we'll have enough time to *iterate* in review; as in, author would respond to reviews and then reviewer would come back for round to. But, I could write a section in the instructor notes for this, with a big "second half of demo time pending!" if we want to allow for that potential.



instructor notes for performing code review Verified 90422c4

Reviewers

jaclyn-taroni
At least 1 approving review is required to merge this pull request.

Still in progress? Convert to draft

Assignees

None—assign yourself

Labels

None yet

Projects

None yet

Sign in now to use Zenhub



GitHub allows us to create multiple issue templates for a repository

We can create Markdown files with some features:

- Structure we find useful (e.g., a checklist or set of headers)
- A YAML header indicating things like labels and prefixes for the title



New analysis example

Use this issue template for filing a new analysis issue

[Get started](#)

Other issue

Use this issue template to describe a current issue with an analysis or documentation (that is not a new example analysis)

[Get started](#)

Report a problem or ask a question

Use this template to report a problem or ask a question about our examples.

[Get started](#)

Don't see your issue here? [Open a blank issue.](#)

[Edit templates](#)

[Sign in now](#) to use Zenhub

jaclyn-taroni Add a template for user reports or questions (#450) 33cdeff · 2 years ago History

Name	Last commit message	Last commit date
..		
new-analysis.md	Add some issue templates (#319)	3 years ago
other-issue.md	Add a template for user reports or questions (#450)	2 years ago
report-a-problem-or-ask-a-question.md	Add a template for user reports or questions (#450)	2 years ago

Customize the issue creation experience with a config.yml file. [Learn more about configuring a template chooser.](#)

Sign in now to use Zenhub


```
---  
name: New analysis example  
about: Use this issue template for filing a new analysis issue  
title: 'New Analysis Example:'  
labels: new analysis example  
assignees: ''  
  
---
```

```
### What are the goals of this new example analysis?
```

```
<!-- What should this new example show that's not currently shown in the existing samples -->  
<!-- What was the background and context that lead to this new example? -->  
<!-- Links to github comments or related issues are also helpful -->
```

```
### What kind of dataset will this need?
```

```
<!-- Is there a particular species, sample size, experimental model, etc. that's needed -->
```

```
### What steps should be included in this analysis?
```

```
<!-- An outline of what steps should probably be taken would be nice if possible -->
```

```
### What packages/methods do you recommend using or looking into for this analysis?
```

```
<!-- Do you have helpful references and resources we should consult for creating this example? -->
```

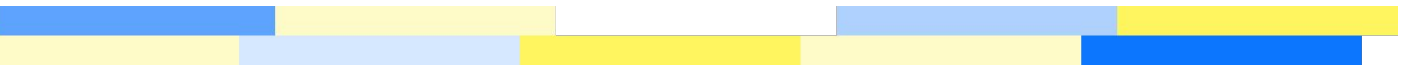
Milestones and projects allow you to further organize your issues

Milestones let you track groups of issues or pull requests and assign a due date. Milestones also let you view the completion percentage and find open and closed issues quickly ([GitHub docs](#)).


Projects can organize issues as a table, spreadsheet, or task-board and let you represent complex relationships between issues with tasklists ([GitHub docs](#)).

We don't use these at the Data Lab; we use something called [ZenHub](#) instead because it allows us to group issues in sprints and represent dependencies between issues (the latter is supposedly coming to GitHub projects in the future!).





What makes a “good” issue?



What makes a “good” issue?

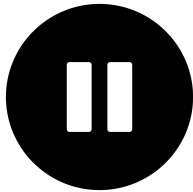


- Title and labels are informative about the contents of the issue “at a glance.”
- Contains enough **context** for someone onboarding to the project or returning to the project after some time to be assigned to contribute to the discussion.
- Communicates **any gotchas** or known unknowns.
- Records ideas about **how to accomplish the task** if the person filing the issue has them.
- Communicates the **requirements for satisfying or closing the issue.**

A good issue is focused

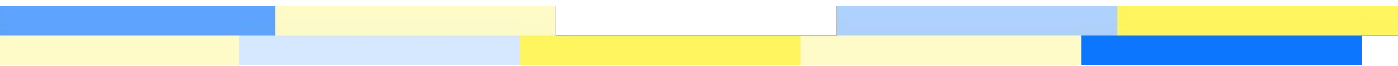


Roughly, the smallest unit of work something could be broken down into to result in a single, functional pull request that is manageable for review.



Some caveats

- Sometimes the author of an issue (or the person that initially comes up with an idea) can't fully flesh out the issue, either because of a time constraint or because some research is required.
 - We often file issues like that prefixed with [PLACEHOLDER] or [WIP] internally.
- Sometimes you need to have a discussion before work can begin. Filing an issue can facilitate an asynchronous discussion and helps make decisions transparent.
 - We often label these with a `discussion` label internally.



Why are focused issues good issues?



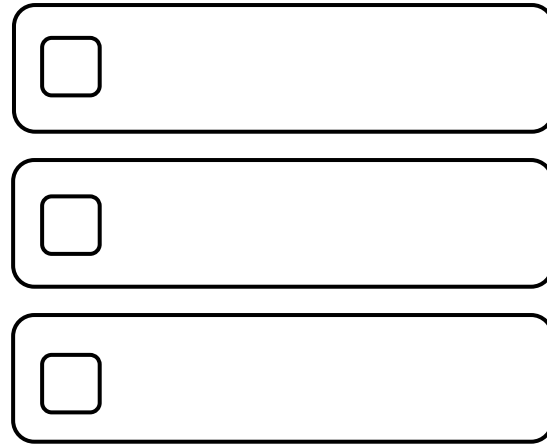
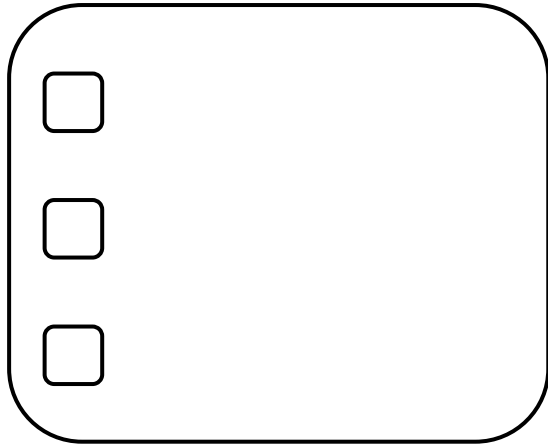
Focused issues set us up for **focused pull requests**



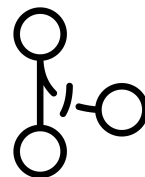
A focused pull request is a more timely and thoroughly reviewed pull request.

Focused issues allow us to spread work across the team

Without duplicating efforts. But that means you need to always assign yourself if you're working on something.



Focused issues might help us **avoid merge conflicts**



If you're combining focused issues with other practices – like planning work as a team and making sure PR branches aren't long-lived – you can assign issues in a manner that minimizes the chances that people will be touching the same parts of the codebase.

But that means you have to stick to what's on the issue!

Focused issues help our team achieve **small wins**



Hitting the closed as completed button feels great!

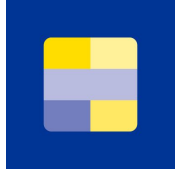
Focused issues let us see the meaningful progress we are making, which can boost the way we feel.

[Amabile and Kramer. *The Power of Small Wins.*](#)



Using issues for planning as a team





Quick tips from Data Lab practices

The Data Lab teams plan their work (i.e., pick issues to tackle) in two week chunks (“sprints”). Here are some things that we’ve learned.

Make sure you **use assignments** on issues. Bonus points if you have an idea of who will review ahead of time! Review is part of everyone’s workload.

You can also fall into a pattern where more folks are reviewing an individual pull request than are necessary and gum up the works that way. Planning ahead could help avoid that, too.

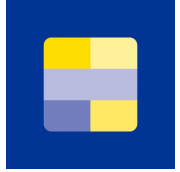


Quick tips from Data Lab practices

The Data Lab teams plan their work (i.e., pick issues to tackle) in two week chunks (“sprints”). Here are some things that we’ve learned.

It can be tempting to – or sometimes necessary to – work on issues that weren’t originally planned during a set time period.

If everything you do will need to be reviewed, unplanned issues creeping in can cause more than one person’s workload to balloon unexpectedly.



Quick tips from Data Lab practices

The Data Lab teams plan their work (i.e., pick issues to tackle) in two week chunks (“sprints”). Here are some things that we’ve learned.

We now have a meeting set up where we nominate issues for technical design discussions. Sometimes an issue isn’t enough and having a synchronous discussion upfront will save both the author and reviewer time.

Folks might end up doubling back and redoing work if there isn’t agreement on approach upfront!

Summary

- GitHub issues let us track and organize our tasks alongside our code, which allows us to connect pull requests to issues
 - A “good issue” contains enough information to facilitate the person directly responsible for carrying out the work getting up to speed quickly
 - Templates can support us in writing good issues
 - Issues that are focused in scope help us create pull requests that are manageable in size and spread work across multiple people on the team when necessary
 - Using issues helps the Data Lab to plan review workloads, in addition to coding workloads, but we’ve needed to combine them with other practices
- 