

How to make a pull request

Imagine you are TyTitan and this is your repository. Your initial code commit (which could be your last lab from last term) is in the *master* branch and the lab 1 code is in a branch named *lab1-validation*.

1. Click on “New Pull Request”

The screenshot shows the GitHub repository page for TyTitan / CS296N. The repository is private. At the top, there are buttons for Watch (2), Star (0), and Fork (0). Below these are tabs for Code, Issues (0), Pull requests (0), Projects (0), Insights, and Settings. The main content area shows 'Lab assignments for CS296N' with an 'Edit' button. Below this, there are statistics: 4 commits, 3 branches, and 0 releases. A dropdown menu for 'Branch: lab1-authentic...' is open, and the 'New pull request' button is circled in red. To the right of the dropdown are buttons for 'Create new file', 'Upload files', 'Find file', and 'Clone or download'. Below the dropdown, it says 'This branch is 1 commit ahead of master.' and provides links for 'Pull request' and 'Compare'. A table of commits is shown, with the latest commit being 'beta version done' by TyTitan 11 minutes ago. The table lists files: BookInfo.Tests, BookInfo, .gitignore, and README.md, along with their commit messages and timestamps.

2. Choose the branch with your lab beta code as the “compare” branch.
Note that the *base* and *compare* branches must contain code that has differences.
3. Select reviewer: the student who is reviewing your code.
4. Add a comment.
5. Click on “Create Pull Request”.

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).

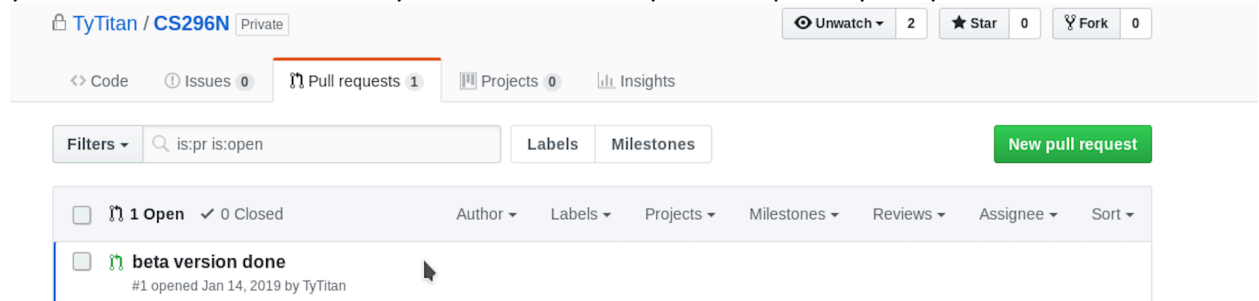
The screenshot shows the GitHub 'Open a pull request' form. At the top, there are dropdowns for 'base: master' and 'compare: lab1-authentication', both circled in red. A red warning message states 'Can't automatically merge. Don't worry, you can still create the pull request.' Below this, there is a text area for the pull request title and description. The title is 'beta version done'. The description is 'My beta version of lab 1 is ready for review.' To the right of the text area, there are sections for 'Reviewers', 'Assignees', 'Labels', 'Projects', and 'Milestone'. The 'Reviewers' section shows 'Ironlenny' selected and circled in red. The 'Assignees' section shows 'No one—assign yourself'. The 'Labels' section shows 'None yet'. The 'Projects' section shows 'None yet'. The 'Milestone' section shows 'No milestone'. At the bottom right, there is a green 'Create pull request' button, which is also circled in red.

The pull request is now sent to your reviewer, ironlenny.

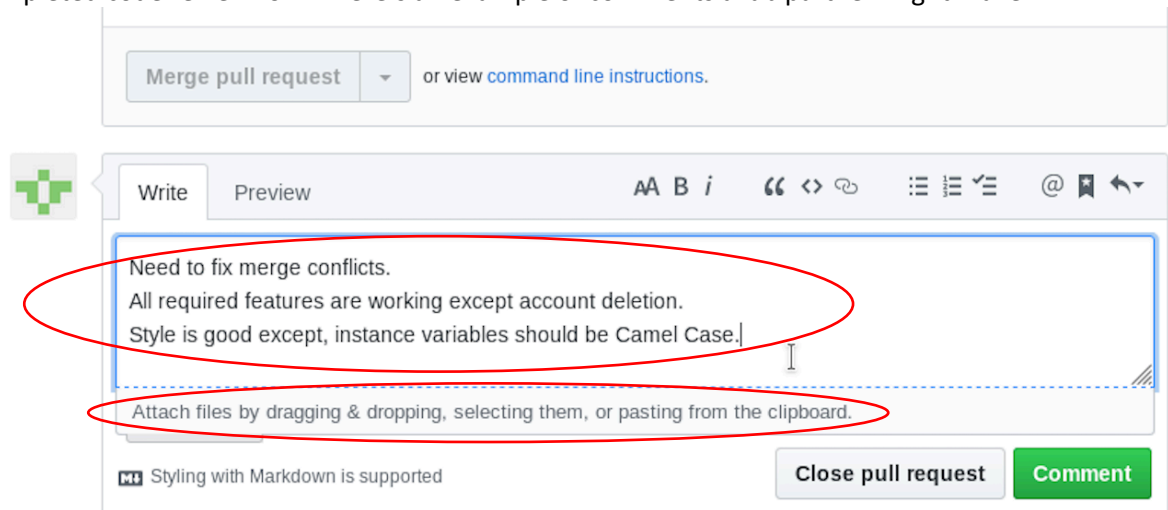
Doing a code review on GitHub

Doing a Code Review

Now imagine you are TyTitan's lab partner, ironlenny. You just got an e-mail with the PR, when you follow the link in the e-mail you will be taken to TyTitan's repository and you'll see this:

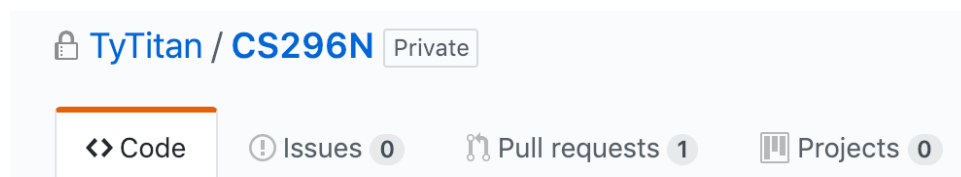


Now you can pull the branch, look at the code, run it, and make comments and attach a completed code review form. Here's an example of comments a lab partner might make:



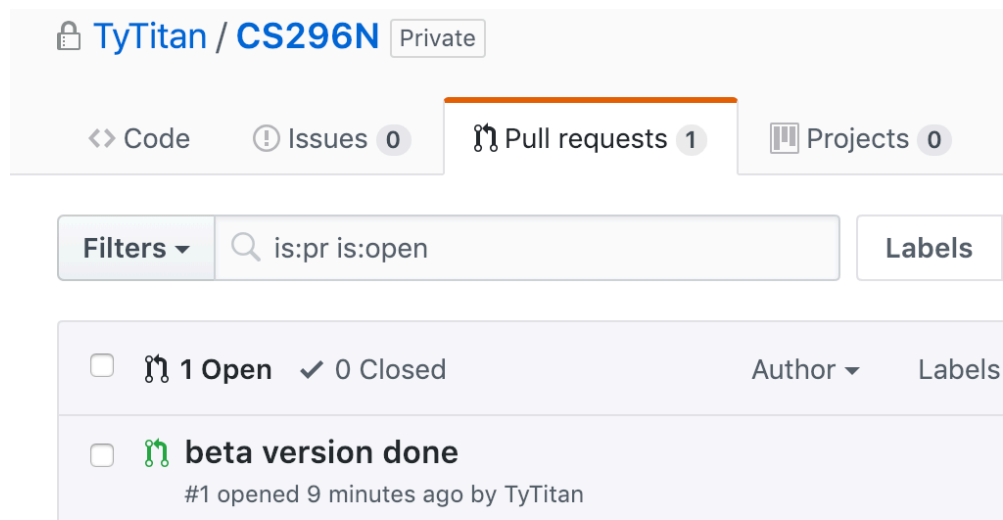
Reading the code review from your lab partner

Now imagine you are TyTitan again. After your lab partner has done the code review, you can read their comments by first clicking on "Pull requests"



Then select the current pull request
(see next page)

Doing a code review on GitHub



TyTitan / CS296N Private

Code Issues 0 Pull requests 1 Projects 0

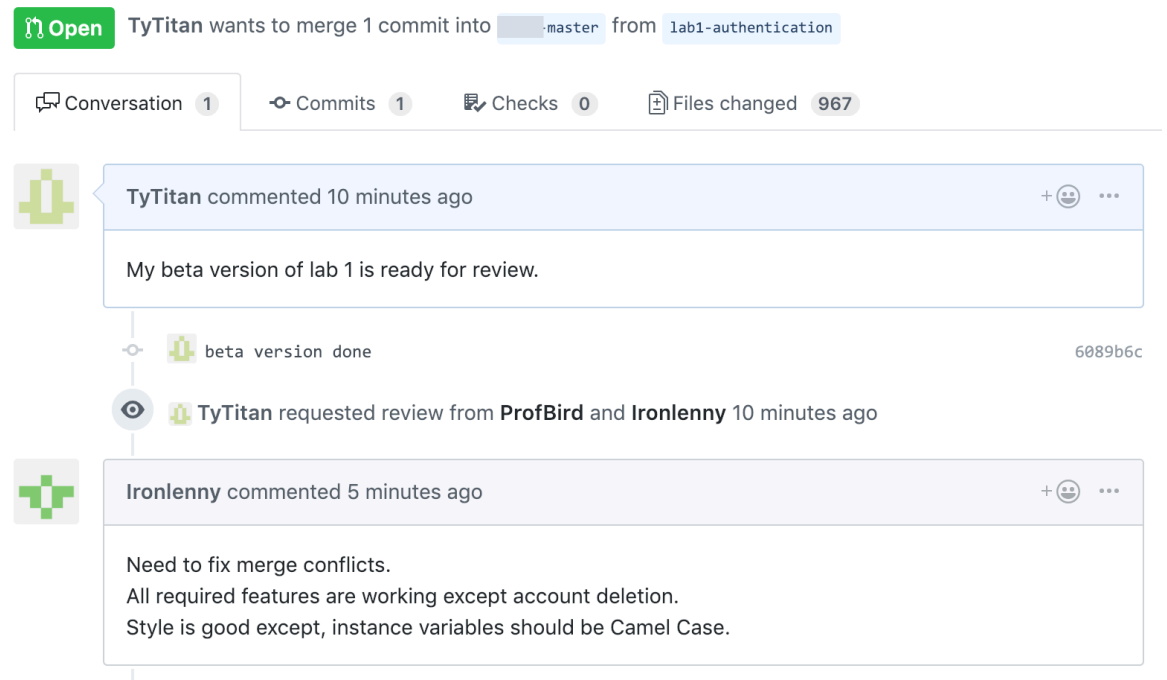
Filters is:pr is:open Labels

1 Open 0 Closed Author Labels

beta version done
#1 opened 9 minutes ago by TyTitan

Then read your lab partner's comments.

beta version done #1



Open TyTitan wants to merge 1 commit into master from lab1-authentication

Conversation 1 Commits 1 Checks 0 Files changed 967

TyTitan commented 10 minutes ago

My beta version of lab 1 is ready for review.

beta version done 6089b6c

TyTitan requested review from ProfBird and Ironlenny 10 minutes ago

Ironlenny commented 5 minutes ago

Need to fix merge conflicts.
All required features are working except account deletion.
Style is good except, instance variables should be Camel Case.

Your lab partner, ironlenny, doesn't need to do the merge or close the PR, you can do that yourself once you are satisfied that the code is production-worthy. Your merge will end up back in master, but keep the Lab1-Validation branch to show what you did for that lab.

It's fine to make multiple changes to the beta branch and get feedback from your lab partner multiple times if your lab partner is willing.