Lab 4: Enforcing Workflows and Code Reviews

Author: Gourav Shah

Publisher: School of Devops

Version : v2024.05.01.01

Here is what you are going to learn in this lab,

- You would begin by defining branch protection rules on GitHub
- Then you would learn about enforcing code reviews via raising pull requests

Enforcing Workflow with Branch Policies

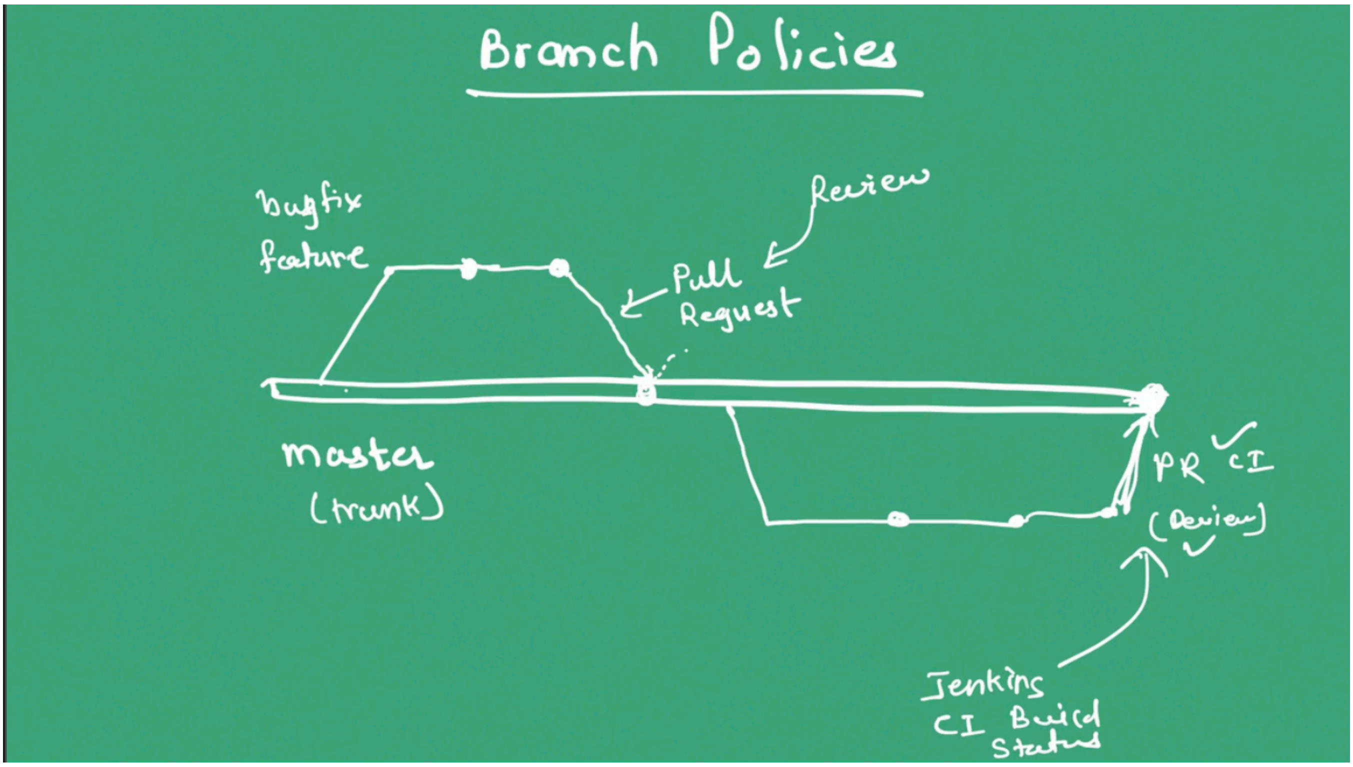
Reading List:

- Web: Guide to Trunk Based Development
- Video: Trunk Based Development Explained

Lets create branch policies based on trunk based development workflow. With trunk based development,

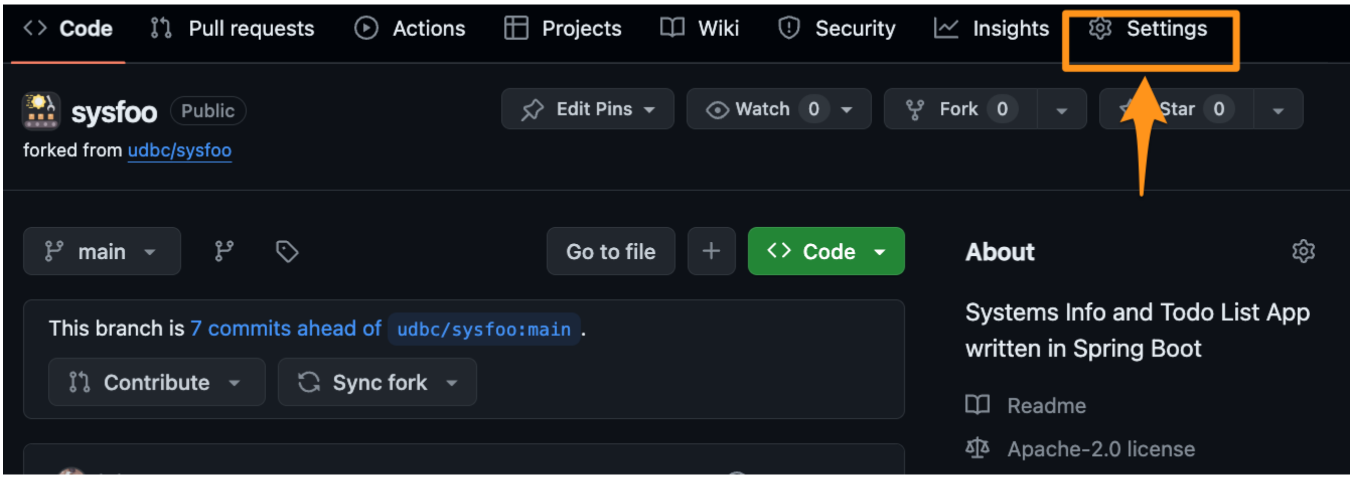
- Master branch is the trunk/main line of code, which is locked. No direct checkins to master allowed.
- Every bugfix gets its own branch
- Every feature gets its own branch too
- Features branches are typically short lived
- Merges to master require pull request
- Further policies could be added to enforce the code review as well as integrate with jenkins ci to run per branch pipeline

This is how it should look like,



To define the branch policies,

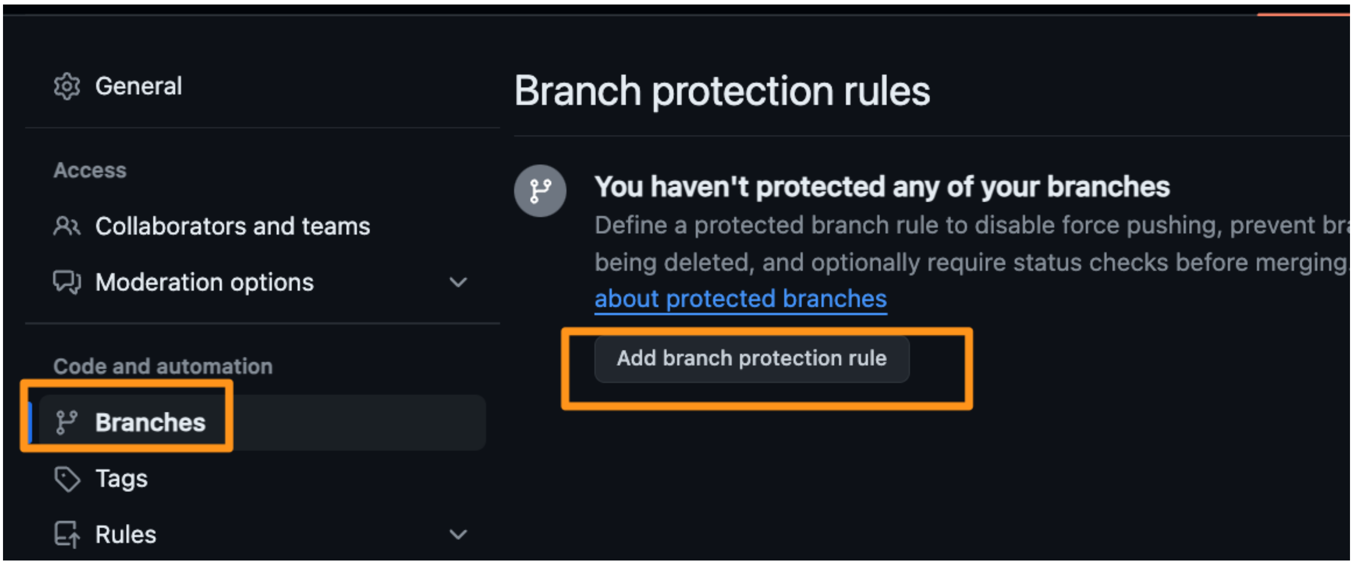
- Goto **settings** for your github repository and from **Branches** select `Add branch protection rule` option



forked from udbc/sysfoo

This branch is 7 commits ahead of `udbc/sysfoo:main` .

Systems Info and Todo List App written in Spring Boot



General

Access

Collaborators and teams

Moderation options

Code and automation

Branches

Tags

Rules

Branch protection rules

You haven't protected any of your branches

Define a protected branch rule to disable force pushing, prevent bra

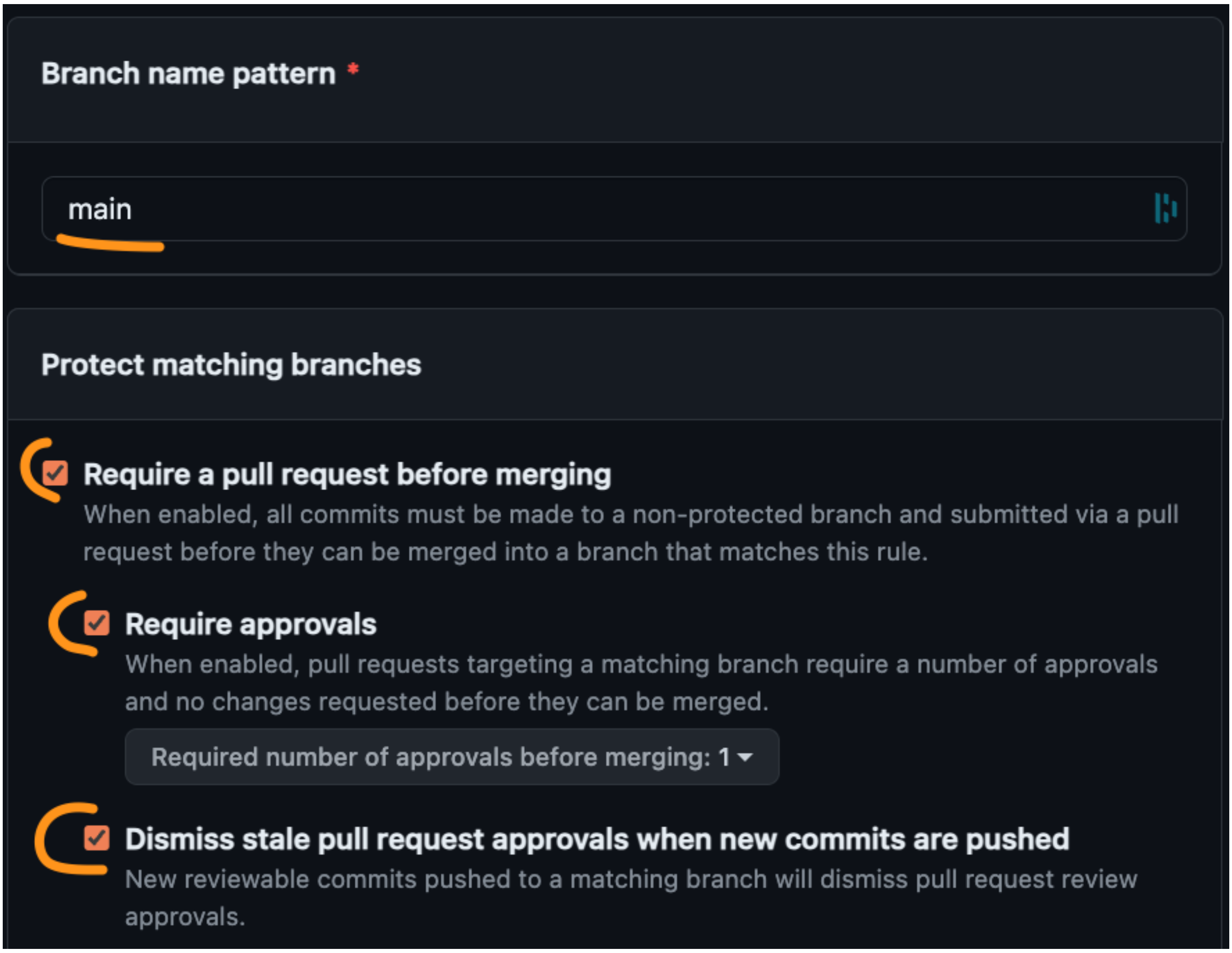
being deleted, and optionally require status checks before merging.

about protected branches

Add branch protection rule

- Add the following set of rules
  - provide branch name as **main**
  - **Require pull request reviews before merging**
    - Require Approvals (select 1 or 2 reviewers)
    - Dismiss stale pull request approvals when new commits are pushed
  - **Require status checks to pass before merging**
    - Select jenkins pipeline job here

The following image depict the branch protection rules



Branch name pattern *

main

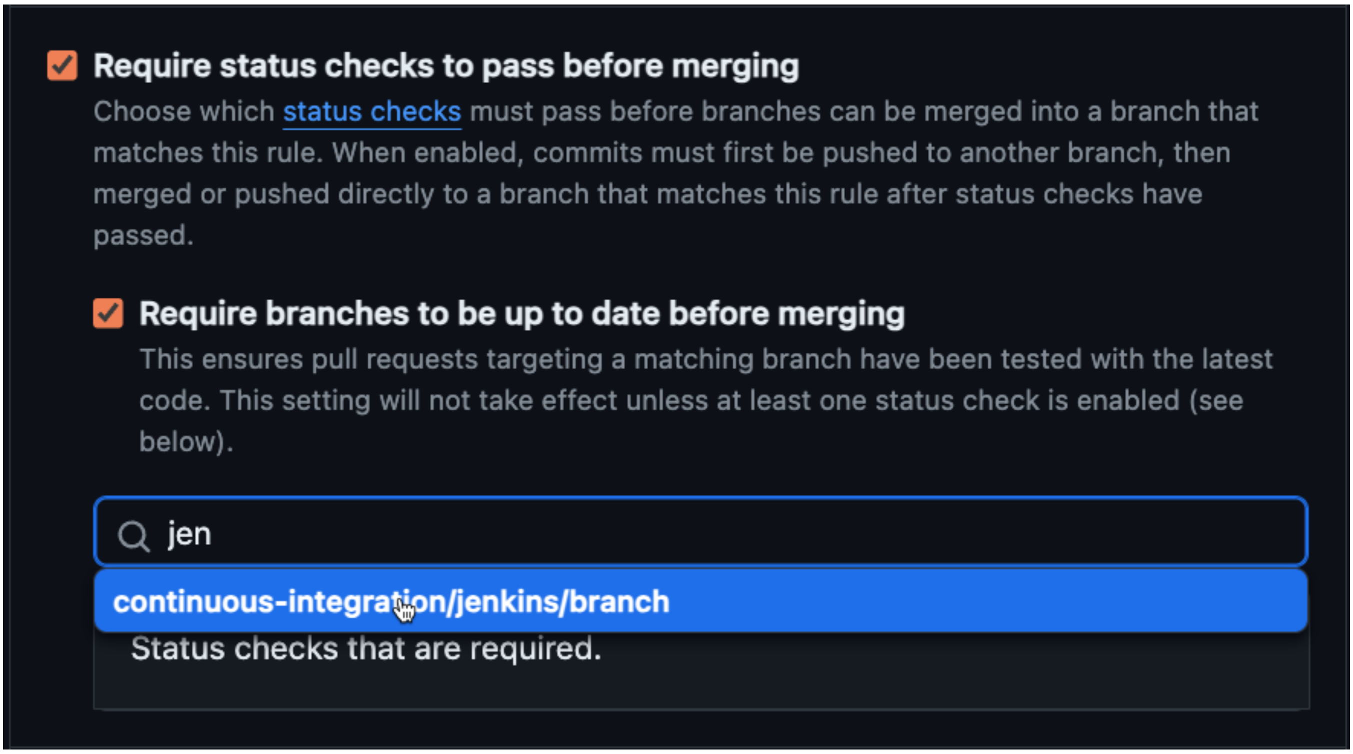
Protect matching branches

☑ **Require a pull request before merging**
When enabled, all commits must be made to a non-protected branch and submitted via a pull request before they can be merged into a branch that matches this rule.

☑ **Require approvals**
When enabled, pull requests targeting a matching branch require a number of approvals and no changes requested before they can be merged.

Required number of approvals before merging: 1 ▾

☑ **Dismiss stale pull request approvals when new commits are pushed**
New reviewable commits pushed to a matching branch will dismiss pull request review approvals.



☑ **Require status checks to pass before merging**
Choose which status checks must pass before branches can be merged into a branch that matches this rule. When enabled, commits must first be pushed to another branch, then merged or pushed directly to a branch that matches this rule after status checks have passed.

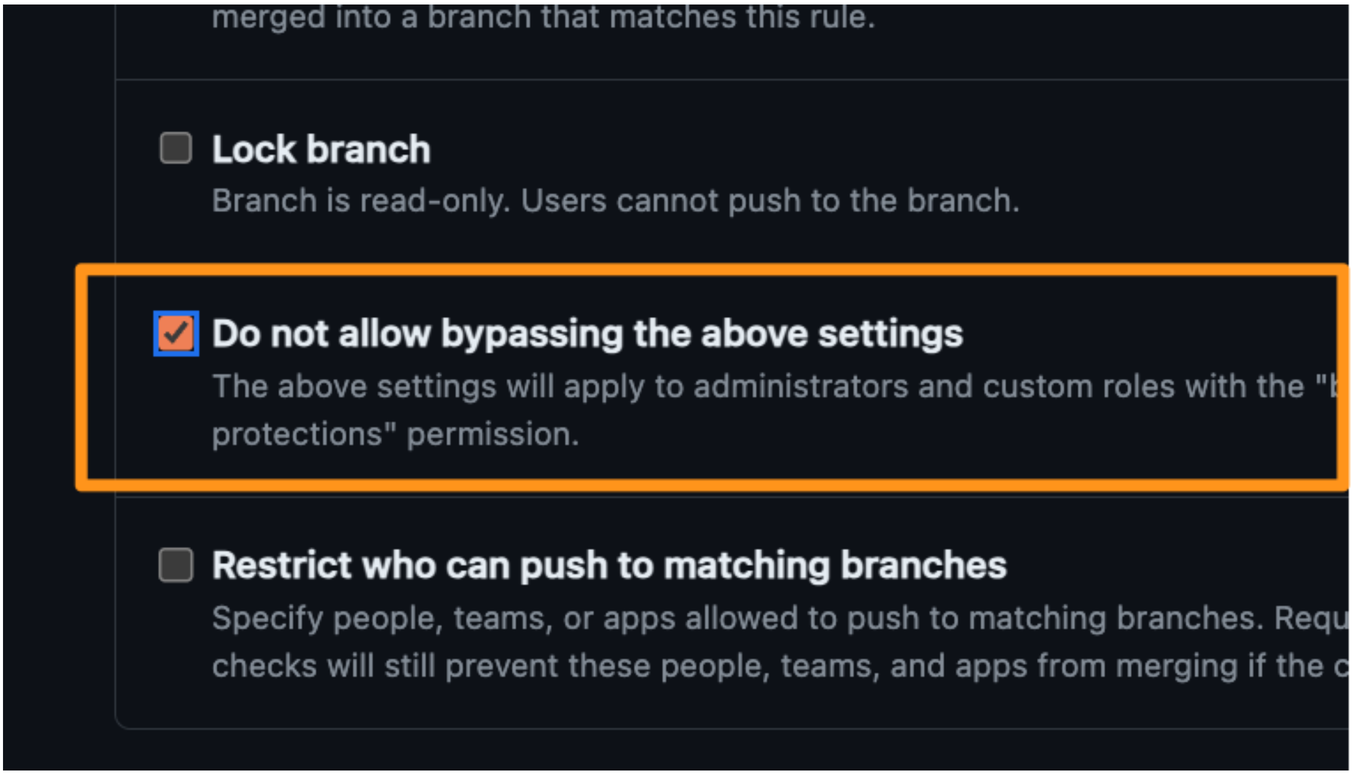
☑ **Require branches to be up to date before merging**
This ensures pull requests targeting a matching branch have been tested with the latest code. This setting will not take effect unless at least one status check is enabled (see below).

jen

continuous-integration/jenkins/branch

Status checks that are required.

Ensure to select Do not allow bypassing the above settings, otherwise you being the admin of this repository, these will not be effective for you.



merged into a branch that matches this rule.

□ **Lock branch**
Branch is read-only. Users cannot push to the branch.

☑ **Do not allow bypassing the above settings**
The above settings will apply to administrators and custom roles with the "b
protections" permission.

□ **Restrict who can push to matching branches**
Specify people, teams, or apps allowed to push to matching branches. Requ
checks will still prevent these people, teams, and apps from merging if the c

Testing Branch Protection Rules

You should have cloned your own fork of *sysfoo* repository locally before you proceed. To test the branch protection rules, from the local repository, make some changes to any file e.g. `test.md`. Try adding it to git and commit those changes as,

```
cd sysfoo/
touch test.md
git status
git add test.md
git commit -m "testing"
git push origin master
```

When you are try to push the changes to a remote, it should throws an error `master protected branch hook declined`. Thats because you can't directly push to master anymore.

You could undo changes by resetting the commit as,

`replace YOURLASTCOMMITID with the actual commit id following`

```
git log
git reset --hard HEAD~1
```

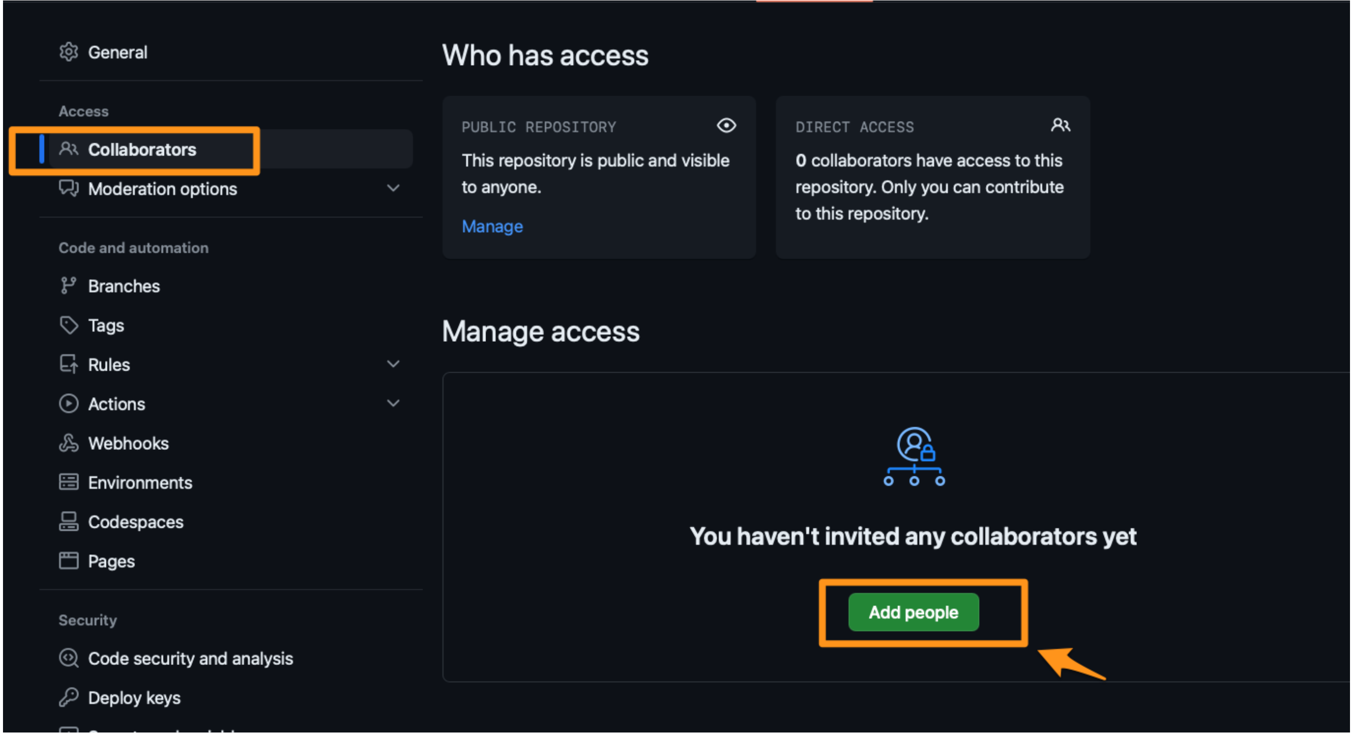
Once you do a hard reset, it will roll back the changes you have made to the commit history and also to the local workspace.

Add a Reviewer to your Repository

In order to get your code reviewed, you need to add a collaborator to your repository first. If you do not have a reviewer to pair up with, just create a new GitHub account and assign that as reviewer.

Steps to add reviewer are as follows,

- From the repository settings, select **Collaborators** => **Add people**



Who has access

PUBLIC REPOSITORY

This repository is public and visible to anyone.

Manage

DIRECT ACCESS

0 collaborators have access to this repository. Only you can contribute to this repository.

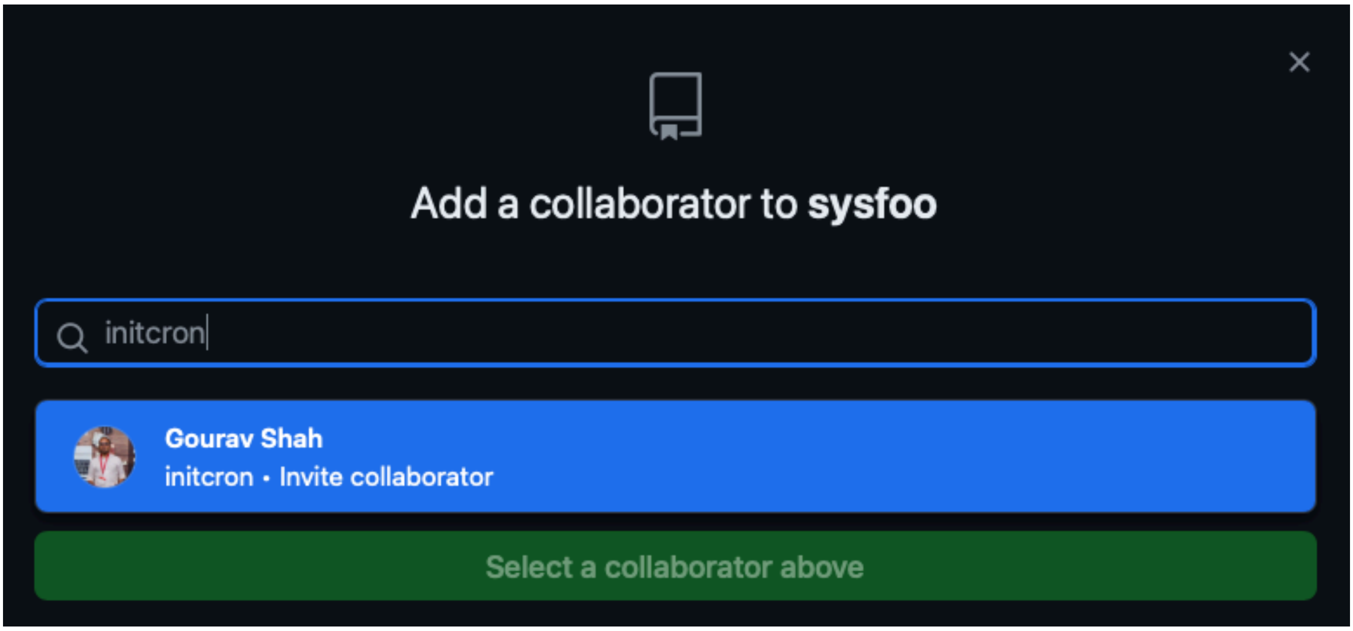
Manage access

You haven't invited any collaborators yet

Add people

- From the manage access page, select the button in green which reads **Invite teams or people**.

- Add a GitHub id of the reviewer and select them as collaborator



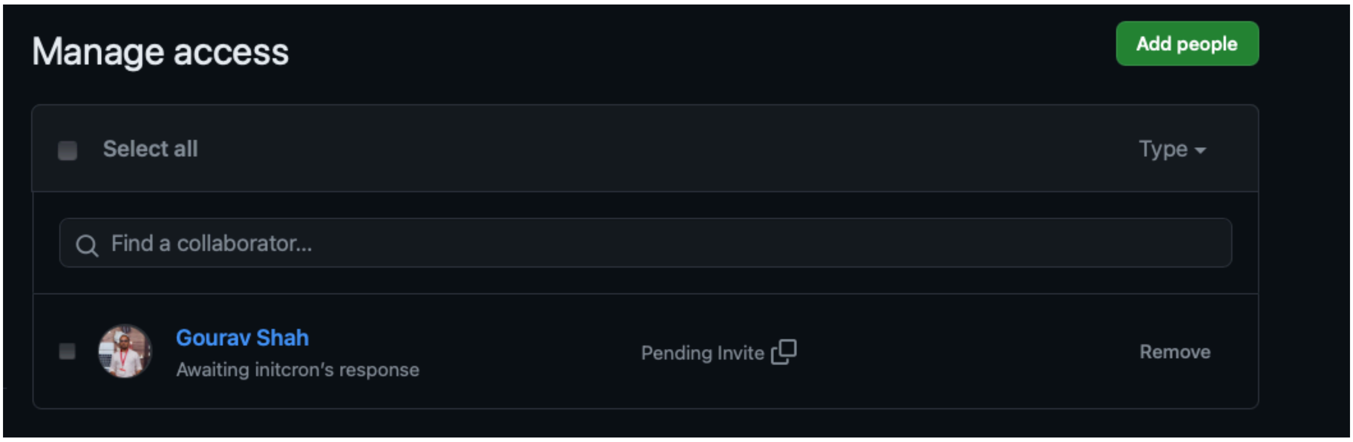
Add a collaborator to **sysfoo**

initcron

Gourav Shah

initcron • Invite collaborator

- If you visit manage access page, it shows the reviewer with **Pending Invite** status. Reviewer should have received an email with the link to accept invitation. Alternately, you could share the link using the icon besides the pending invitation status and share it with reviewer.



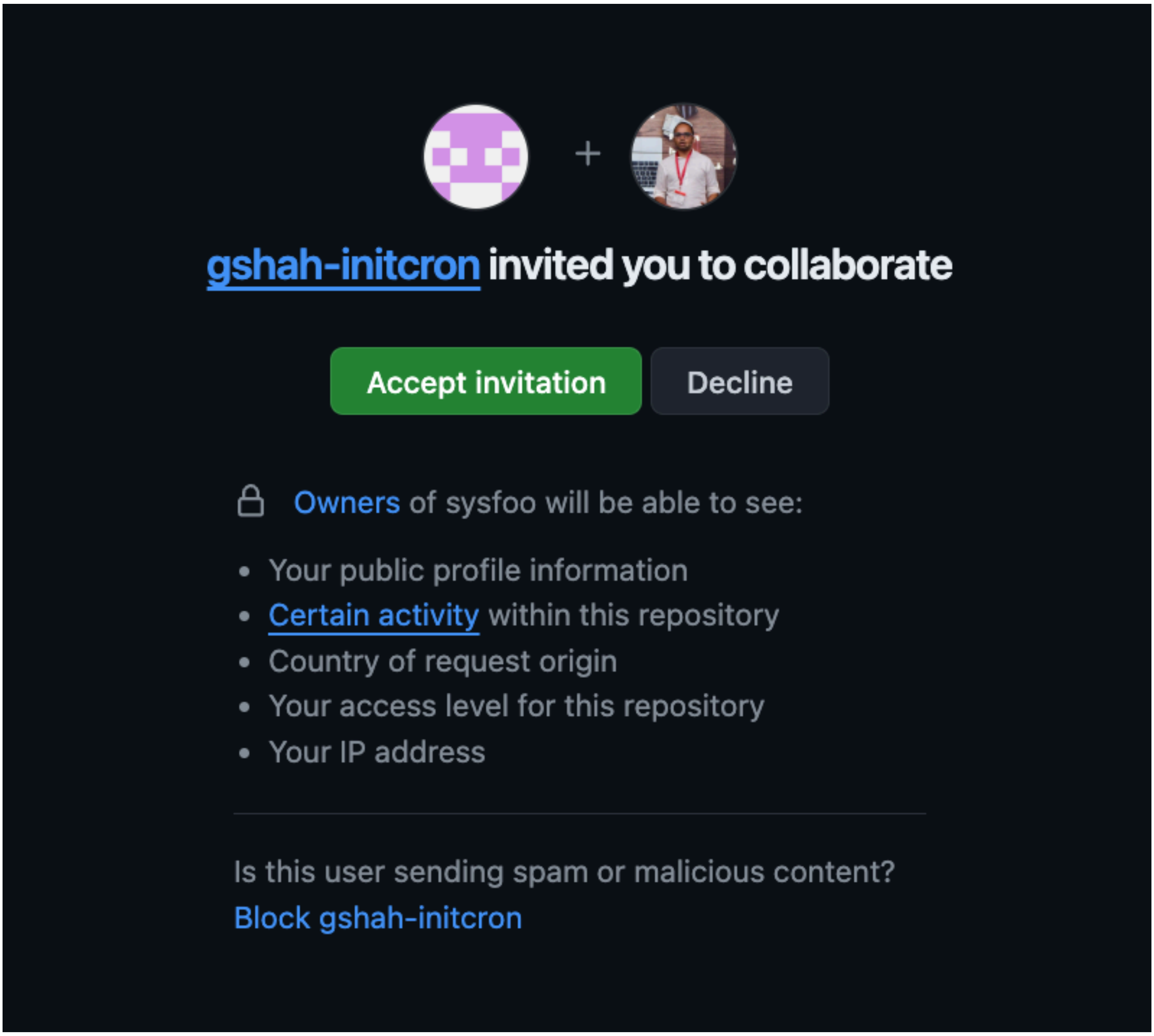
Manage access

Gourav Shah

Awaiting initcron's response

Pending Invite

- Once invitation link is clicked by the reviewer, he/she is shown a page similar to below with **Accept invitation** button in green.



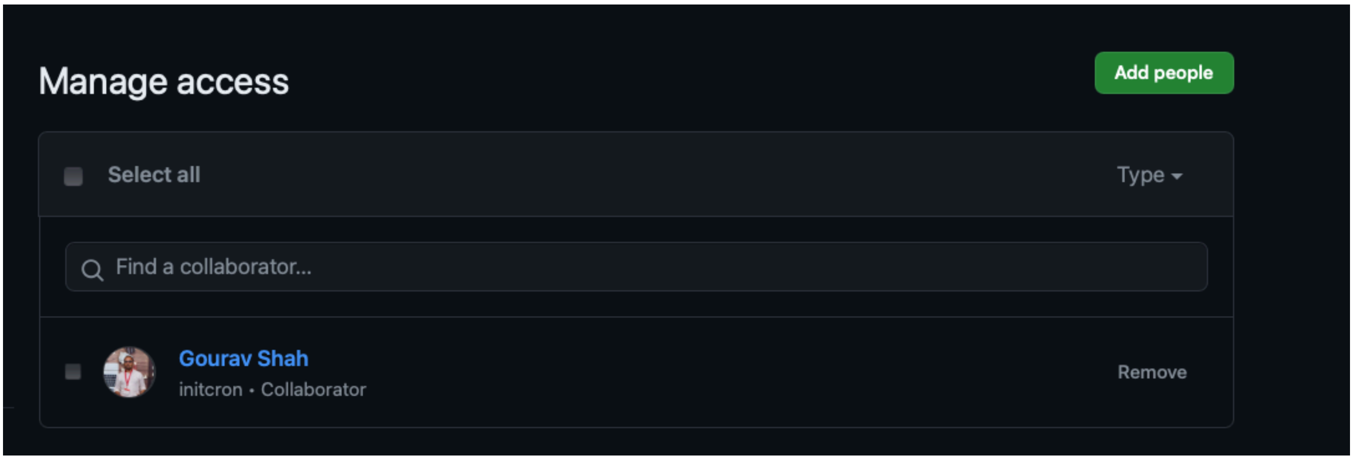
gshah-initcron invited you to collaborate

Owners of sysfoo will be able to see:

- Your public profile information
- Certain activity within this repository
- Country of request origin
- Your access level for this repository
- Your IP address

Is this user sending spam or malicious content?
Block gshah-initcron

- Once invitation is accepted, visiting/refreshing manage access button shows the reviewer with pending invite status gone.



Manage access

Gourav Shah
initcron • Collaborator

A reviewer is successfully added to your account.

Code Reviews with Pull Requests

With this sub section, you will learn how to make changes in master branch by using code reviews and pull requests. You'r task this time is just to update the existing README.md file to the project and incorporate those changes to the trunk i.e. master branch.

- Create a branch `readme` using following command,

```
   git checkout -b readme
```

Running this command, git will create and switch to the *readme* branch.

- Update the `test.md` file by adding a description of the app.

  e.g.

  `filename: test.md`

```
Example App
=========

This is a sample app.
```

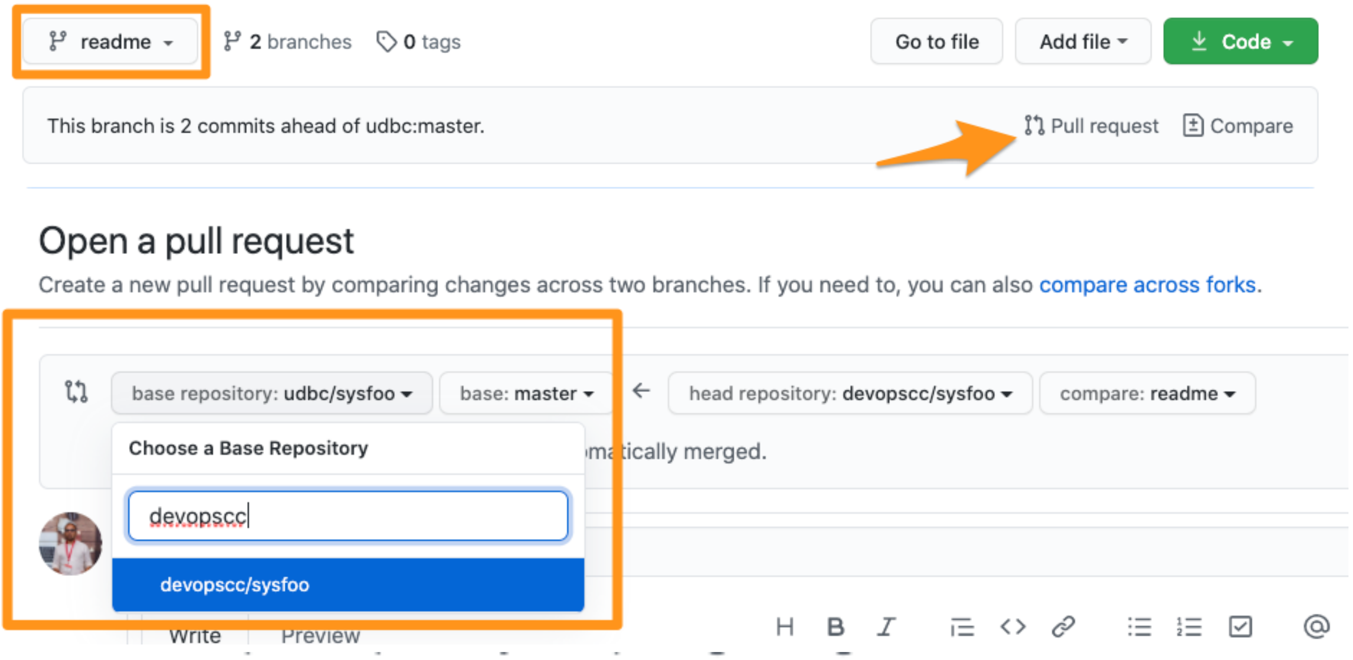
- Save and commit the changes you have made.

```
 git status
 git add test.md
 git commit -m "added application info"
 git push origin readme
```

This pushes the changes to `readme` branch of the repo on GitHub.

- Now go to the github repository and choose branch `readme`, there you should see `new pull request`. The pull request is to help you to merge those changes from the feature branch to master, to the upstream repo you forked, or even to a completely different fork.

- Select `new pull request` and choose your base as master branch, add the description and create pull request.

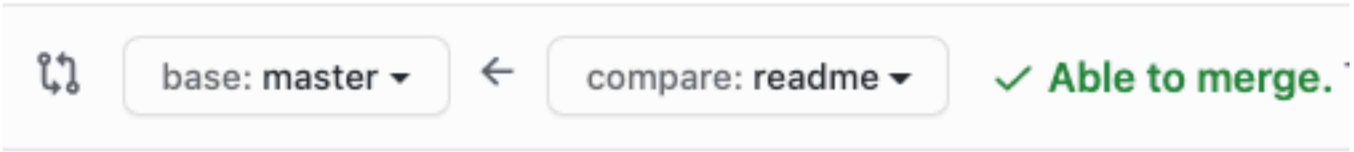


This branch is 2 commits ahead of udbc:master.

Open a pull request

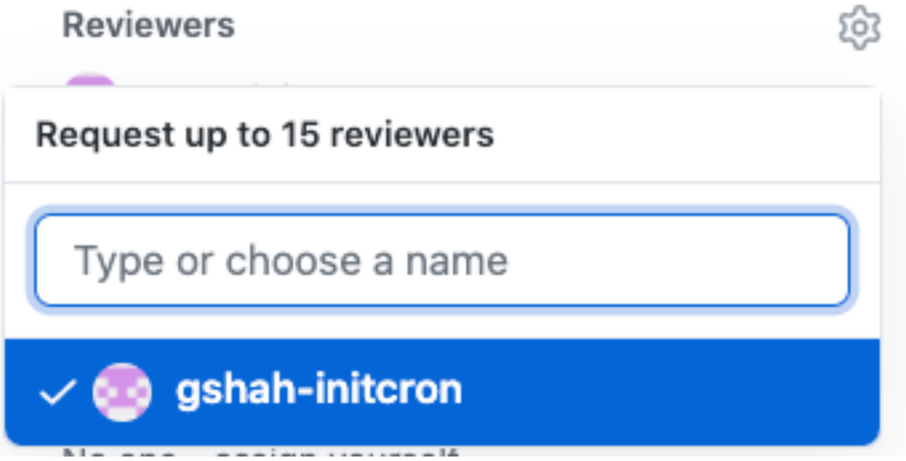
Create a new pull request by comparing changes across two branches. If you need to, you can also compare across forks.

…matically merged.

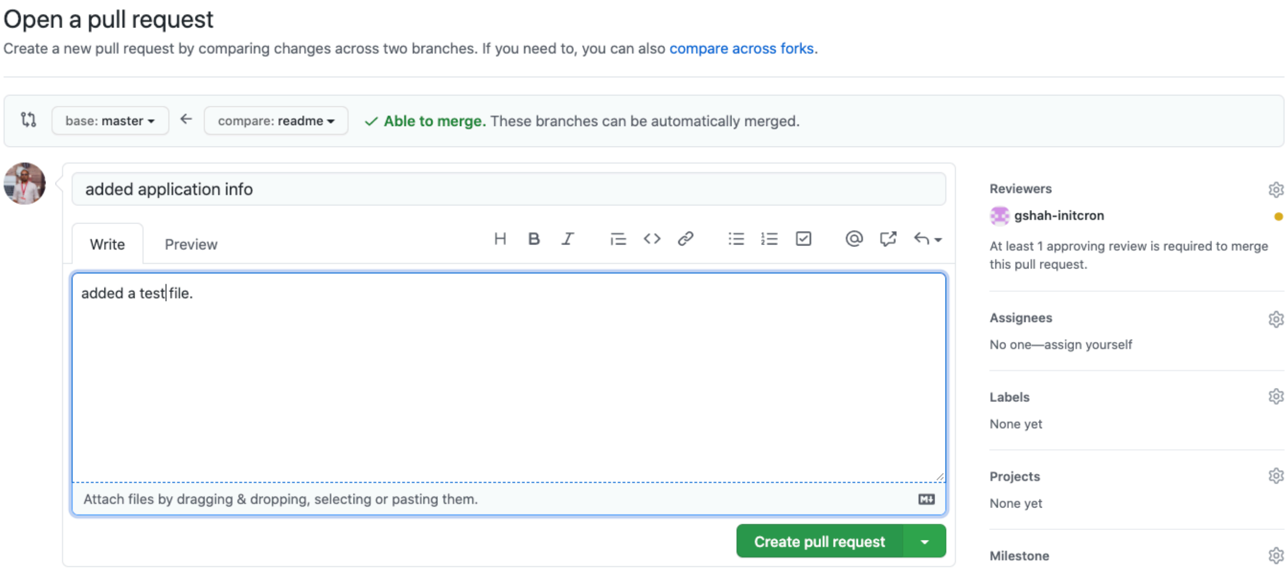


be sure to choose your own repository’s master branch using the drop down menu while choosing the base branch to raise the pull request against.

- Add a reviewer: Creating a pull request does not allow you to merge immediately. It would demand a code review which is been mandated as part of the branch protection rules you have set. Add at least one of your collaborators as a reviewer. If you do not have a collaborator, create another account on GitHub, configure it as a collaborator, and add it as a reviewer.



- Add a description of changes, and check on **Create pull request** button.



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.

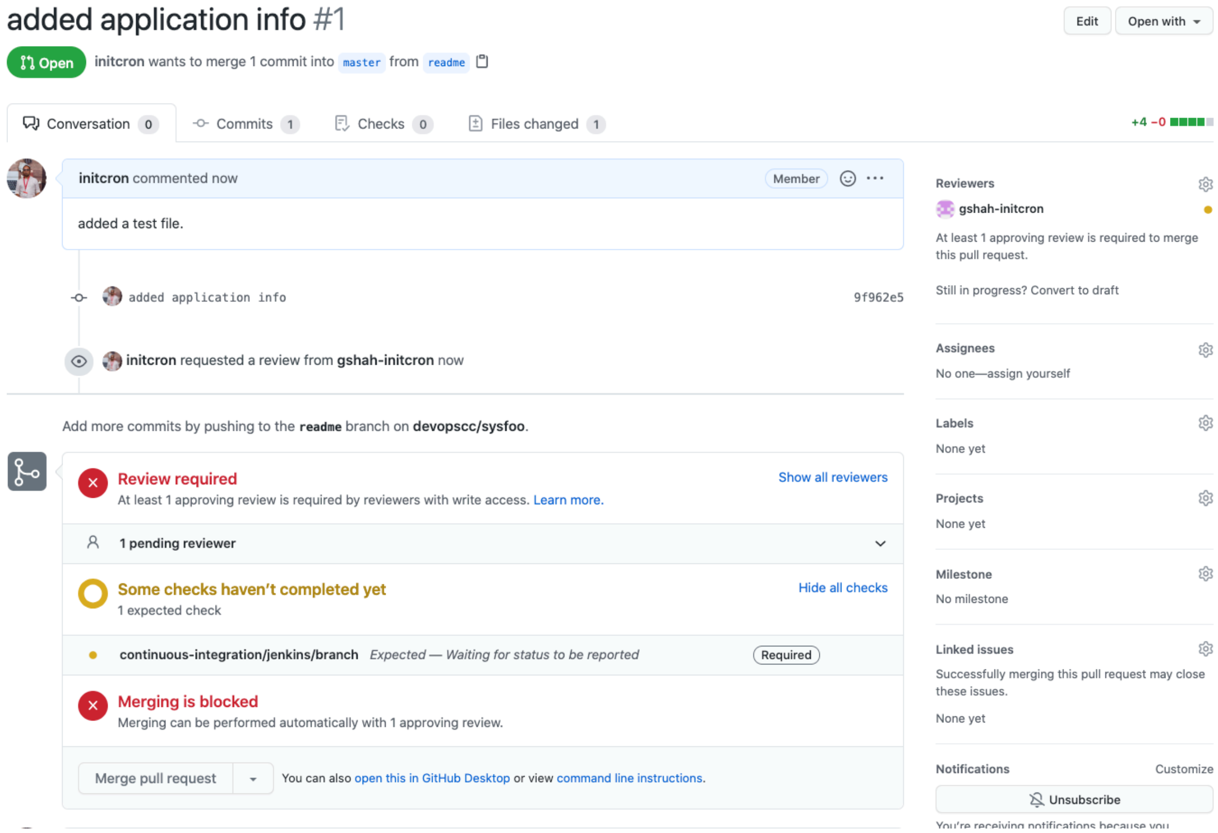
base: master ← compare: readme ✓ **Able to merge.** These branches can be automatically merged.

added application info

added a test file.

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

- If the branch protection rules are in place, at this time you should see the merging being blocked waiting for checks (Jenkins Pipeline) as well as Code Review.



added application info #1

Open · initcron wants to merge 1 commit into `master` from `readme`

initcron commented now

added a test file.

`added application info` 9f962e5

initcron requested a review from gshah-initcron now

Add more commits by pushing to the **readme** branch on **devopscc/sysfoo**.

Review required
At least 1 approving review is required by reviewers with write access. Learn more.

1 pending reviewer

Some checks haven't completed yet
1 expected check

continuous-integration/jenkins/branch *Expected — Waiting for status to be reported* Required

Merging is blocked
Merging can be performed automatically with 1 approving review.

You can also open this in GitHub Desktop or view command line instructions.

Reviewers: gshah-initcron

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

Still in progress? Convert to draft

Assignees: No one—assign yourself

Labels: None yet

Projects: None yet

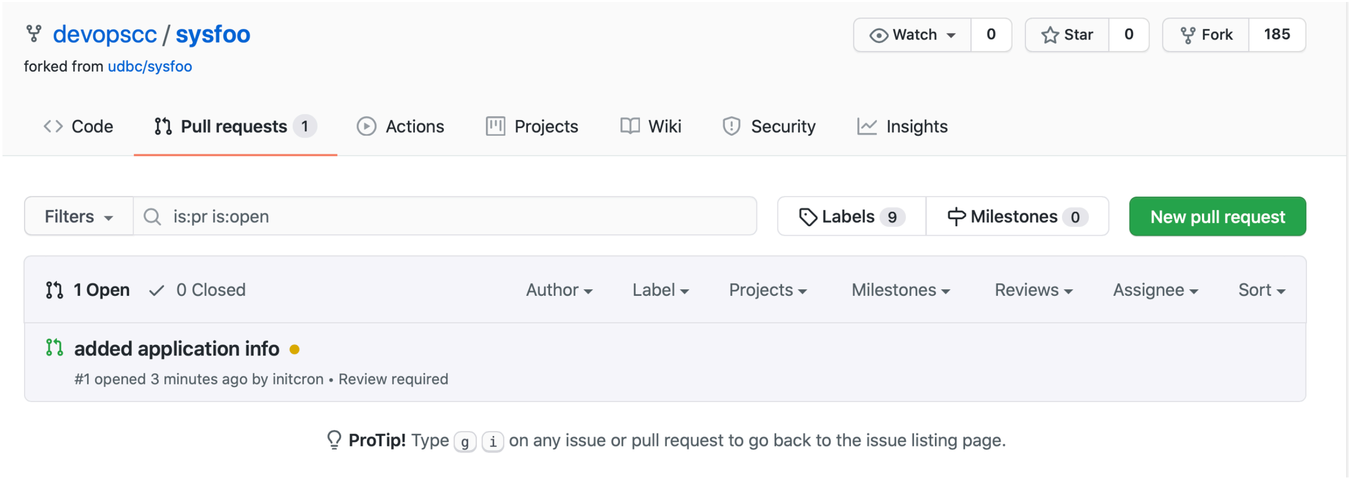
Milestone: No milestone

Linked issues: Successfully merging this pull request may close these issues.

None yet

You're receiving notifications because you

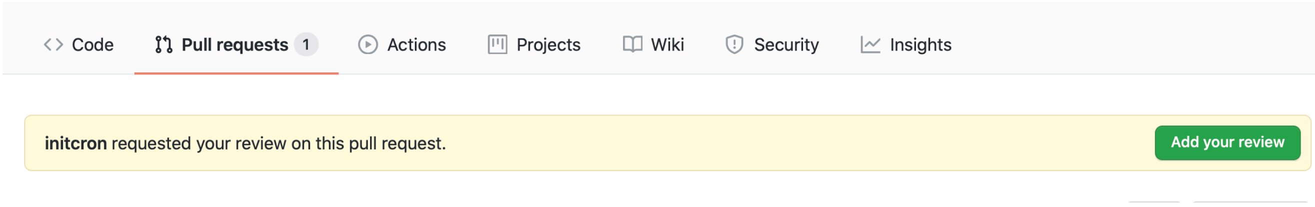
- Submit the Code Review. If you are the reviewer, go to pull requests, review the changes and approve.



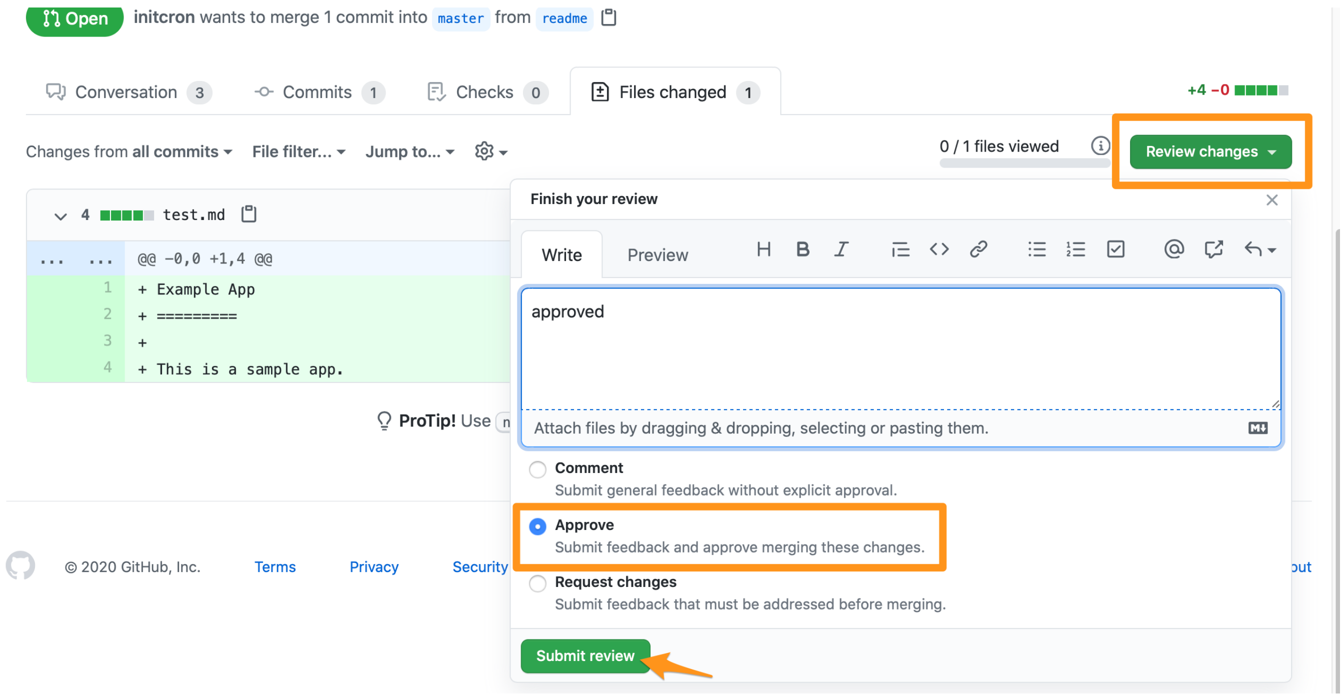
devopscc / sysfoo
forked from udbc/sysfoo

added application info
#1 opened 3 minutes ago by initcron • Review required

ProTip! Type g i on any issue or pull request to go back to the issue listing page.



initcron requested your review on this pull request.



Open **initcron** wants to merge 1 commit into `master` from `readme`

Conversation 3 | Commits 1 | Checks 0 | Files changed 1 | +4 −0

Changes from **all commits** ▾ File filter... ▾ Jump to... ▾ — 0 / 1 files viewed — Review changes ▾

4 test.md

```
@@ -0,0 +1,4 @@
+ Example App
+ =========
+
+ This is a sample app.
```

ProTip! Use n

Finish your review

Write | Preview

approved

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

○ **Comment**
Submit general feedback without explicit approval.

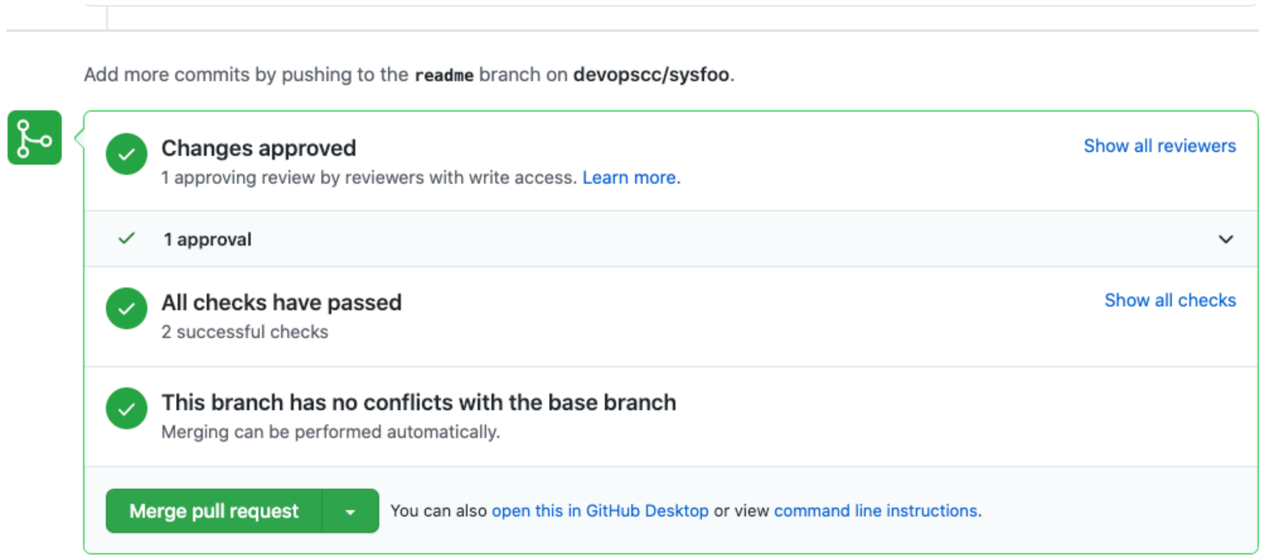
● **Approve**
Submit feedback and approve merging these changes.

○ **Request changes**
Submit feedback that must be addressed before merging.

Submit review

© 2020 GitHub, Inc. Terms Privacy Security

- After approval you could merge the pull request to master. Additionally, you could delete the branch after merge.



Add more commits by pushing to the **readme** branch on **devopscc/sysfoo**.

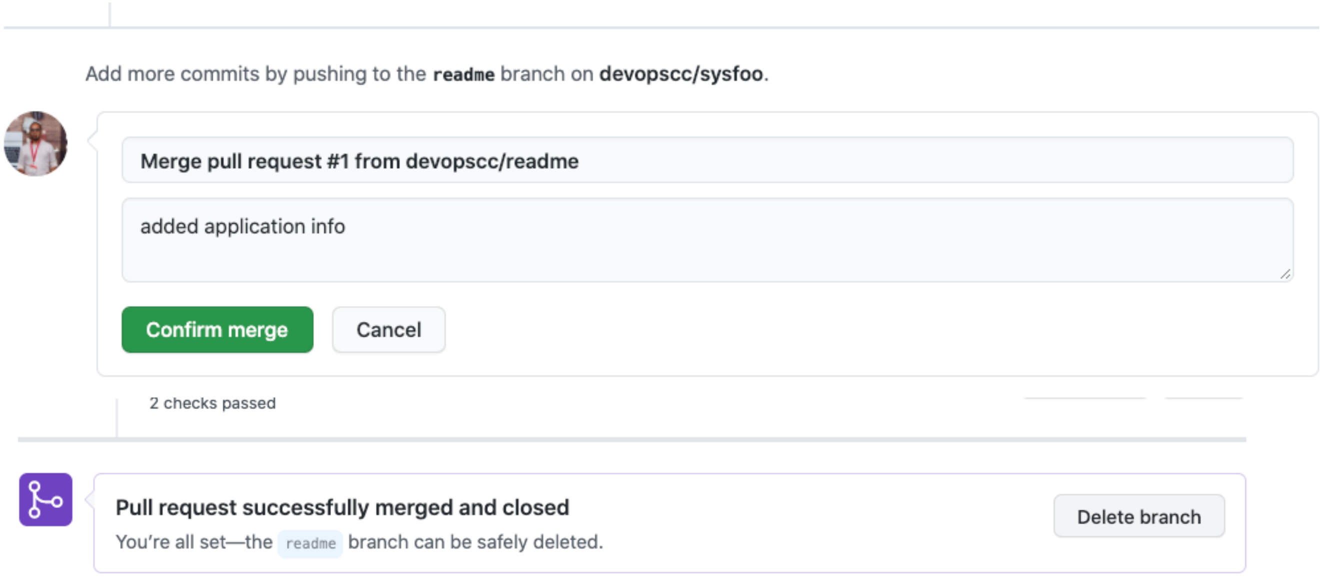
✓ **Changes approved** — Show all reviewers
1 approving review by reviewers with write access. Learn more.

✓ **1 approval**

✓ **All checks have passed** — Show all checks
2 successful checks

✓ **This branch has no conflicts with the base branch**
Merging can be performed automatically.

Merge pull request ▾ You can also open this in GitHub Desktop or view command line instructions.



Add more commits by pushing to the `readme` branch on **devopscc/sysfoo**.

Merge pull request #1 from devopscc/readme

added application info

Pull request successfully merged and closed

You're all set—the `readme` branch can be safely deleted.

This deletes the branch from the remote (GitHub). To delete the branch locally use the following commands,

```
git branch
git checkout master
git branch
git branch -D readme
git branch
```

#cicd/labsv3