Requiring Approval for Pull Requests
Keillor Cogswell

Introduction

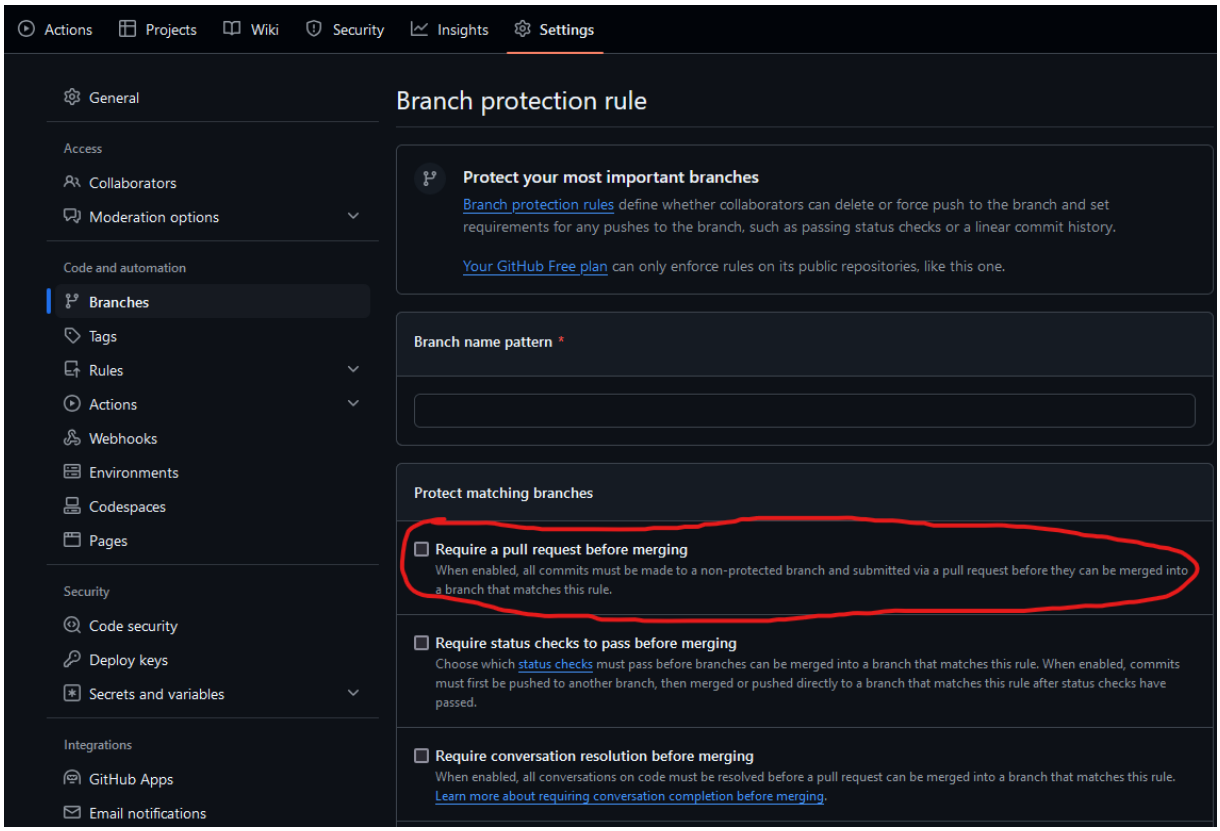
Code changes in branch-based development can be a temperamental matter at times. Perhaps Bobby forgot to unit test his code before pushing it from his feature branch, or something got deleted or overwritten that should not have been. Good source control practices can mitigate these problems, and GitHub is replete with many tools to facilitate these practices.

One way that GitHub helps with source control is by allowing repository owners to enable protections for certain branches. These can range from disabling forced pushes to a branch, to requiring verified signatures or successful deployments, or even locking a branch from further pushes. For the purposes of this tutorial, however, the focus will stay on one key form of protection: requiring pull requests before merging. This requires that any developer making commits must first do so to a non-protected branch before making a pull request for the protected branch. If the pull request is approved, changes will be added to the protected branch as normal.

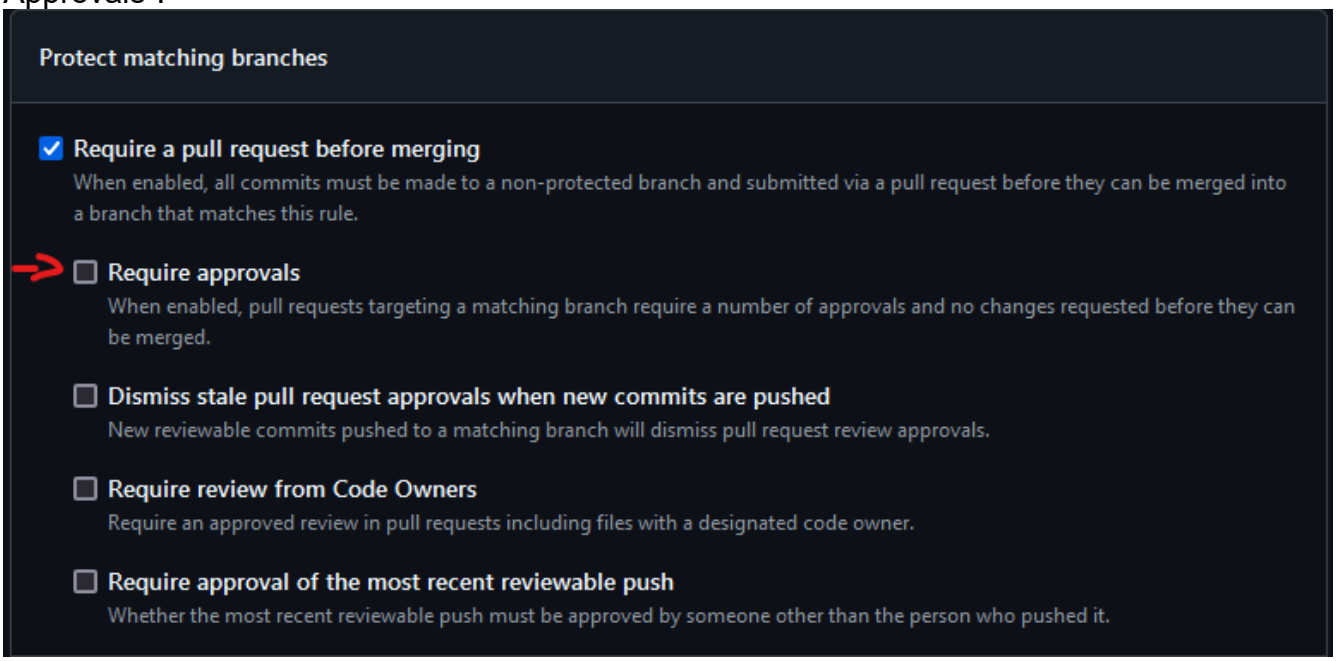
Going a step further, protected branches requiring pull requests before merging can also be set to require reviews or approval for each pull request, and the process of doing so is detailed below.

How to Require Approval

Requiring approval for a protected branch before merging pull requests is a simple process that can be carried out quickly. First, the owner will need to navigate to Settings and go to Branches. There, they can enable the setting to require a pull request before merging.

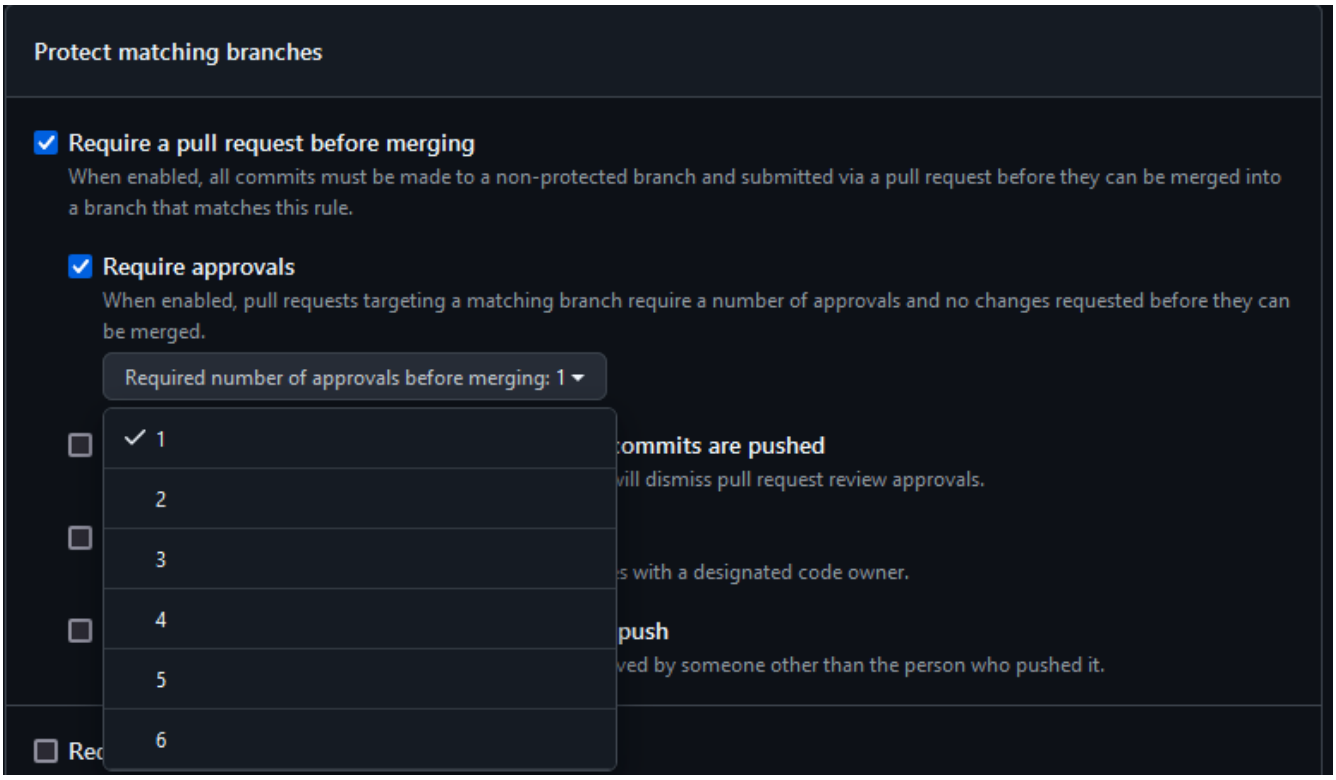


Once this setting is enabled for the branch, a sub-menu will appear below the setting, providing different options. The owner should select the topmost option, that being “Require Approvals”.



The final step involves the owner selecting how many approvals will be required for a pull request. By default, only one approval is needed, but up to six approvals can be required for

each pull request. Team size and structure can dictate the exact number of approvals needed, and it can be modified as needed.



From there, it is simply a matter of configuring these settings for each branch that requires protection. Once the owner is satisfied, any subsequent pull requests to the protected branch will be blocked until enough reviewers approve the changes.

A Word of Caution

While requiring approval for pull requests can add an additional layer of protection for the most important branches of a repository, it can also slow down the implementation of changes due to the added oversight needed for pull requests. Additionally, having specific people designated for review can delay changes if one or more reviewers are unable to perform their duties. These practices may not even be pertinent for some development teams, depending on what source control and code review practices are in place.

When requiring two or more approvals, each reviewer should take care not to shirk their review duties, just because someone else is also looking at the code. Having multiple pairs of eyes on the same code is done so that the development team can gleam a more complete understanding of current code defects. Hence, if a reviewer defers their duties to others without prior agreement, it is less likely that all of the relevant faults in the code will be found.