Vets.gov

GitHub Content Process

Table of Contents

General Tips

1. Don't work on the same file in the same branch at the same time as someone else

2. Don't create new branches from a non-master branch

3. Always make sure you're making updates to the master branch

GitHub Intro

Finding a repository

1. Bookmark the Department of Veterans Affairs account: https://github.com/department-of-veterans-affairs

2. In the repository search bar, start typing to filter to the repo you want to open, and when it appears in the list, click it
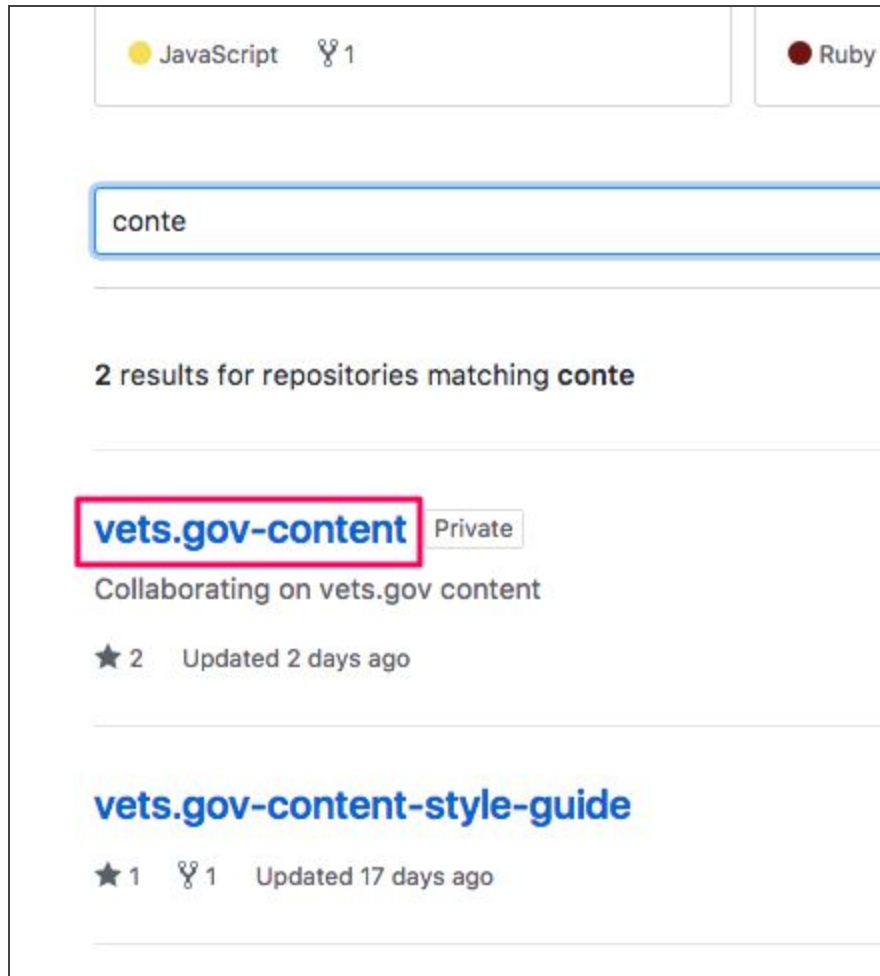
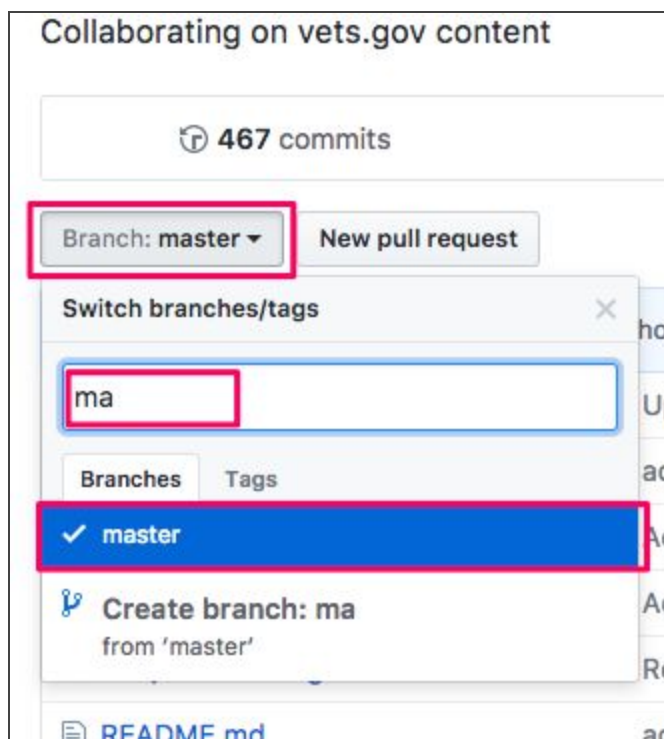3. You can also bookmark the repository directly if you want to come back to it easily: https://github.com/department-of-veterans-affairs/vets.gov-content
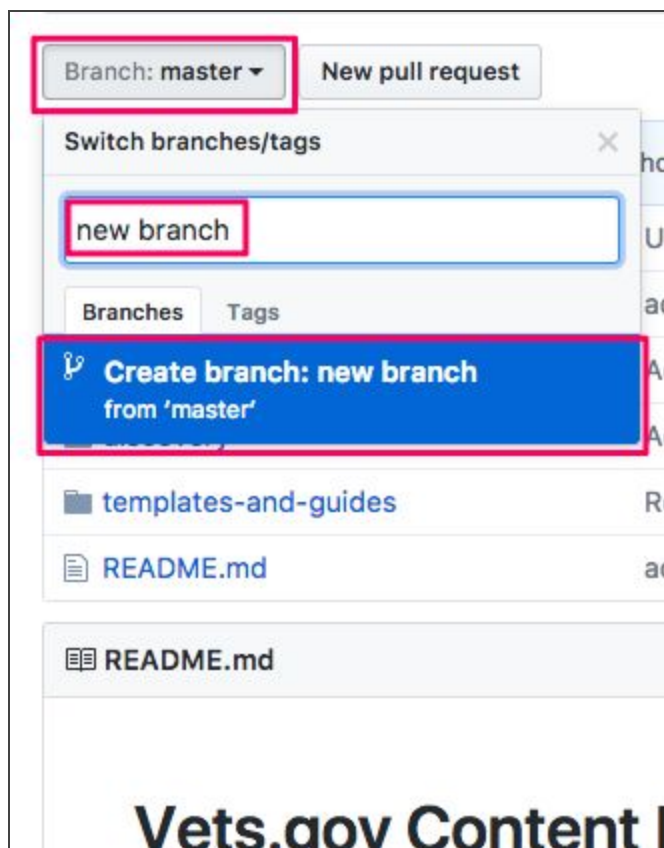
Creating a branch

1. Open the repository in question
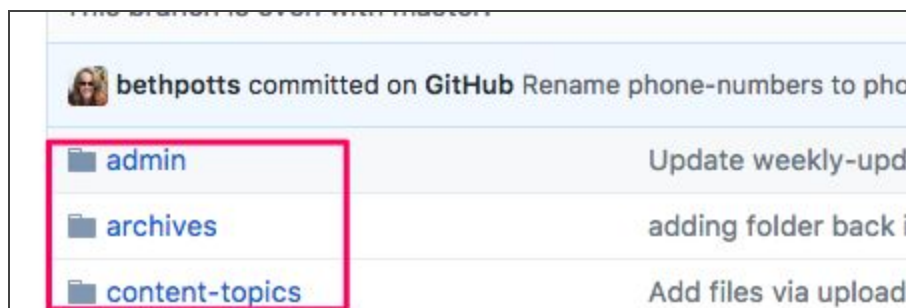
2. Make sure you're on the Code tab
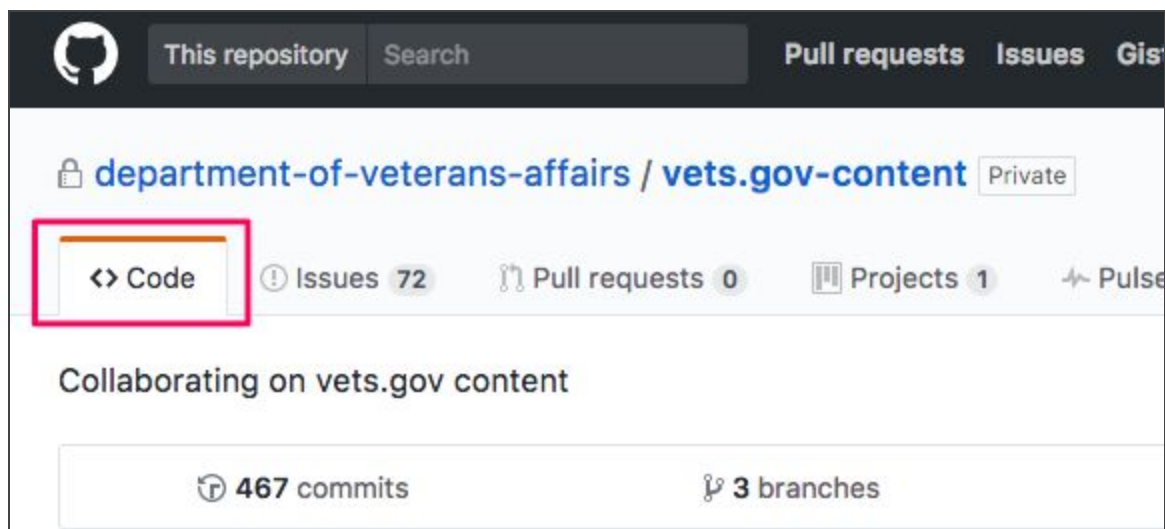
3. Make sure you currently have the master branch selected

4. Once you select master as your base branch, click the branch dropdown again, type the name of your new branch, and select Create branch

5. Now you've created a new branch that's a copy of the master branch, you can click through the files in this new branch to update them (and eventually merge your updates back to master once they're ready to be published)
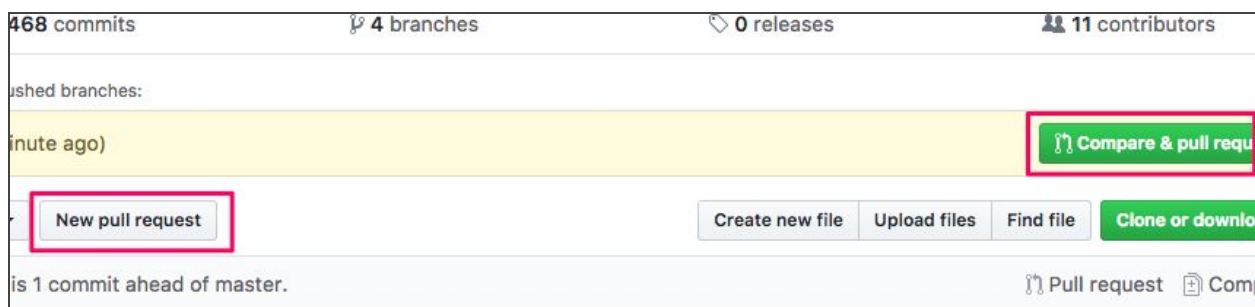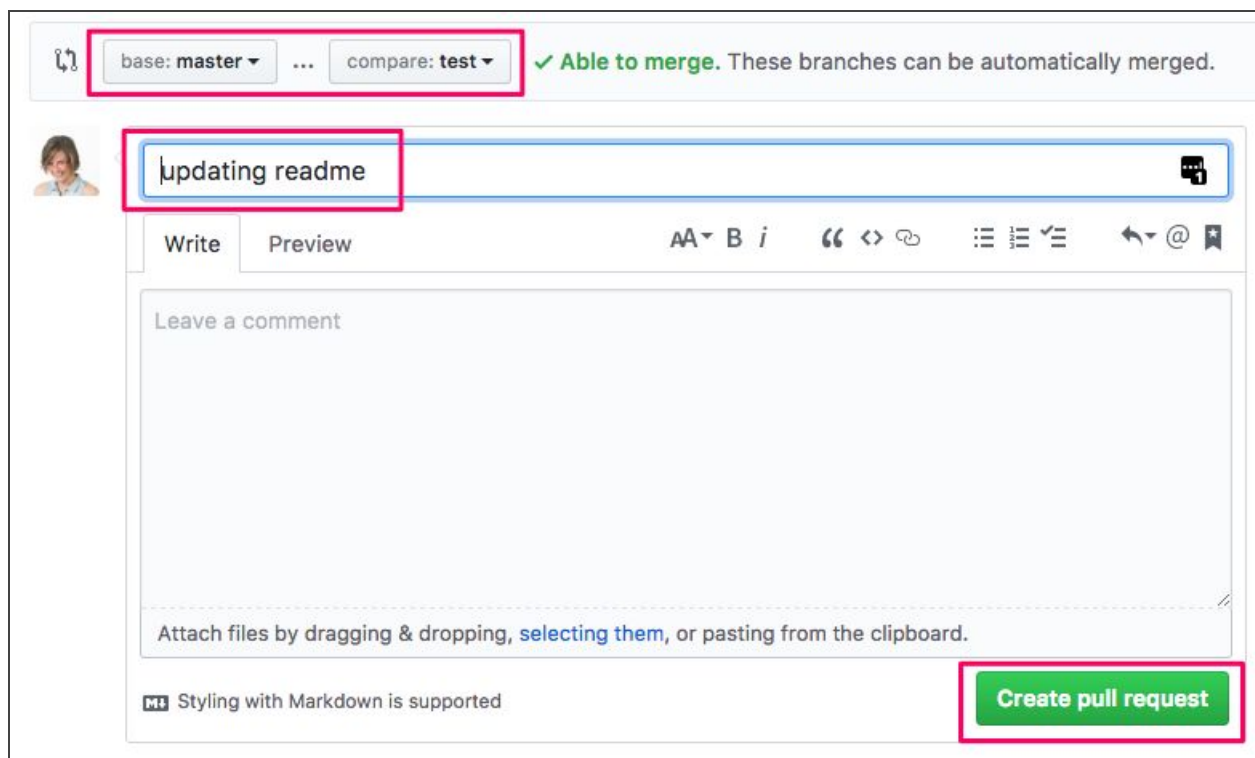

Creating a pull request

1. Once you've made some updates to your branch, and you're ready to merge those changes to master, you create a PR

2. Click on the Code tab

3. Make sure the branch is set to your new one
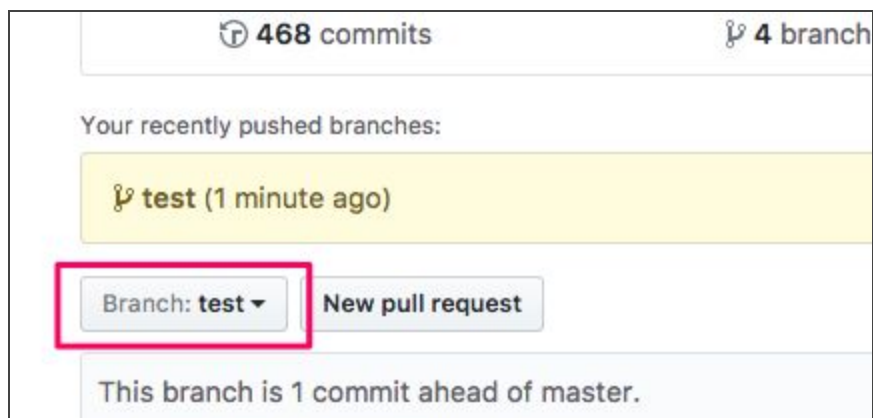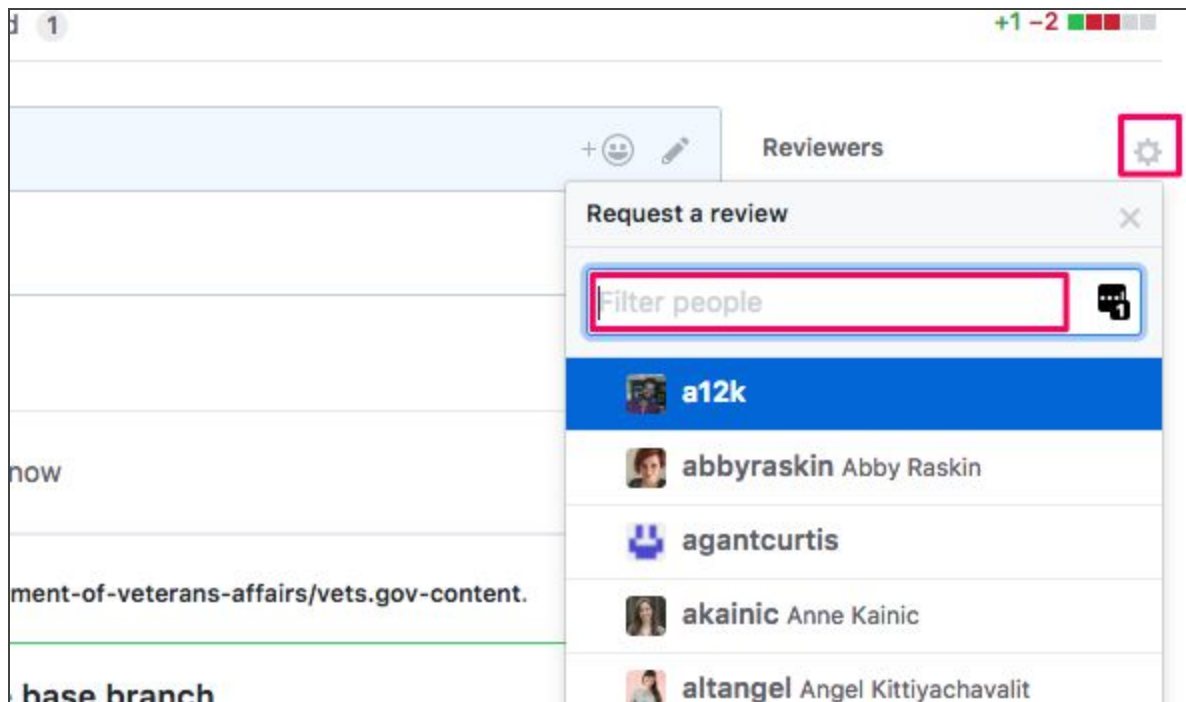
4. Click either of the PR options

5. Check that the base is master, and you're comparing your new branch, enter a title, and click Create Pull Request
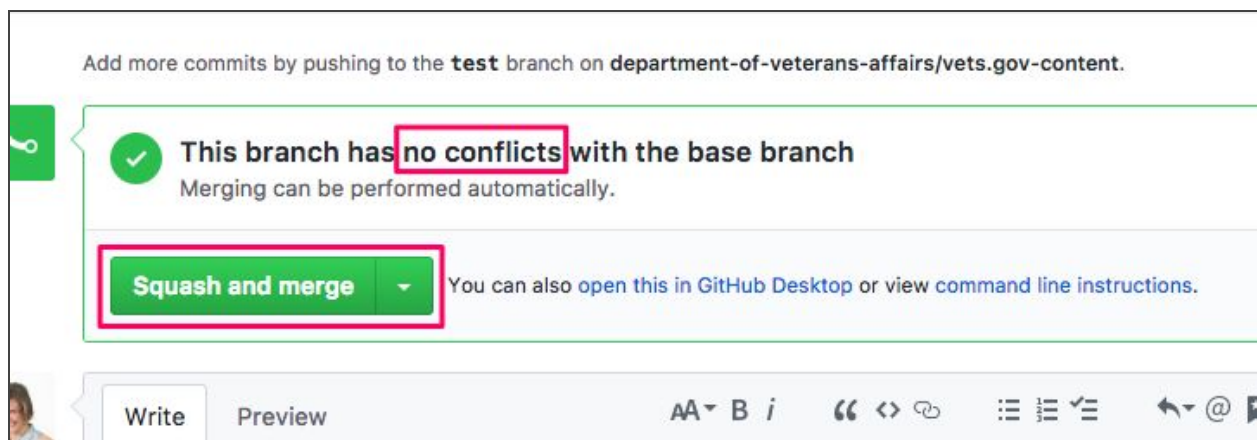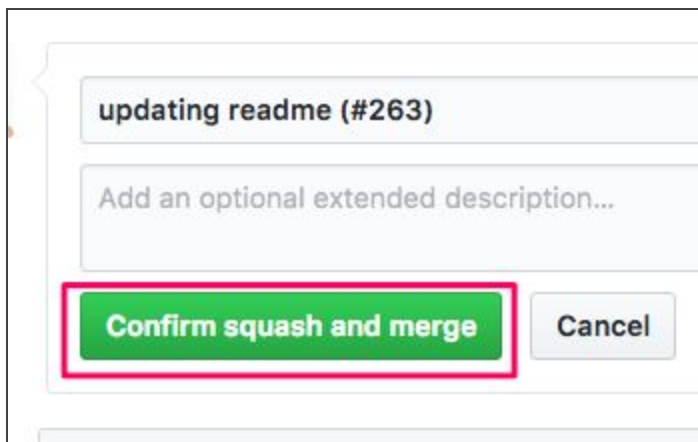
Merging a pull request

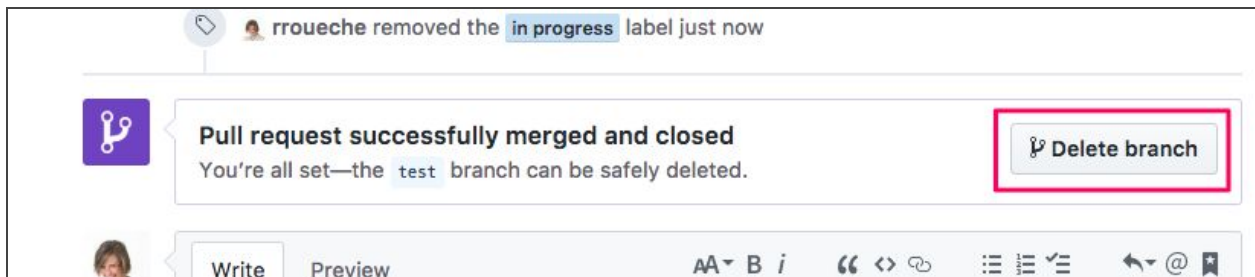1. Get your PR reviewed by selecting other users

2. Once others have approved your changes, and your branch has no conflicts, click Squash and Merge

3. Add any additional notes and click Confirm

4. Delete the old branch


Vets.gov Staging Process

1. Review IA for new topic with IA and content leads (currently Mikki and Amy)

2. Once approved, create issue with the IA, page titles, and breadcrumbs listed out, and assign to Ad Hoc (currently Rachael) to stub out blank pages

3. Each new topic gets new branch off of master with name: [WIP] *topic name*

4. Ad Hoc creates branch and PR and assigns PR to content manager (currently Beth) to coordinate pouring in content

5. PR creation triggers Heroku staging link to be built, where you can see staged content in browser

Vets.gov Editing Process

1. Go to vets-website repo:
   https://github.com/department-of-veterans-affairs/vets-website

2. **Open branch for topic you want to edit** (this red bc you need to make sure you're NOT in master)
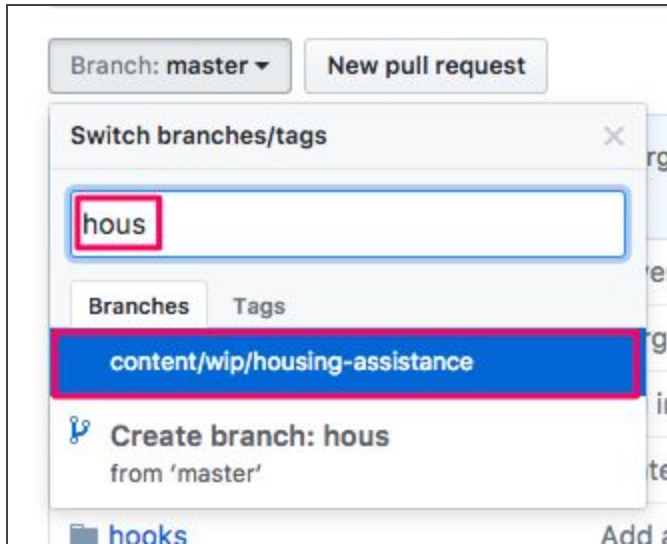
3. In the list of files, go to content > pages > *desired file*
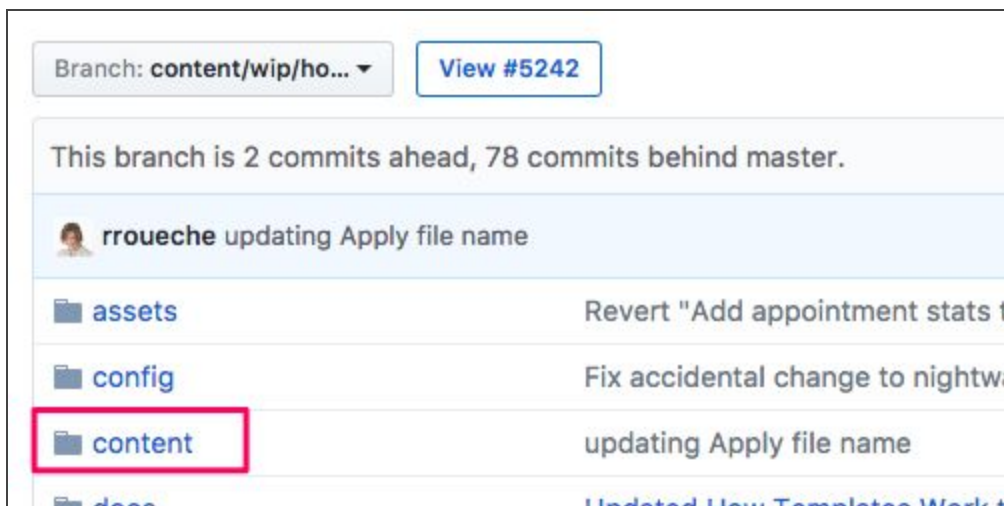
Branch: content/wip/ho... ▾   **vets-website** / **content** /

This branch is 2 commits ahead, 78 commits behind master

👤 **rroueche** updating Apply file name

..

📁 includes   Removes the Manage Benefits button fr

📁 layouts   Removed .section.one, .section.two, .se

📁 **pages**   updating Apply file name

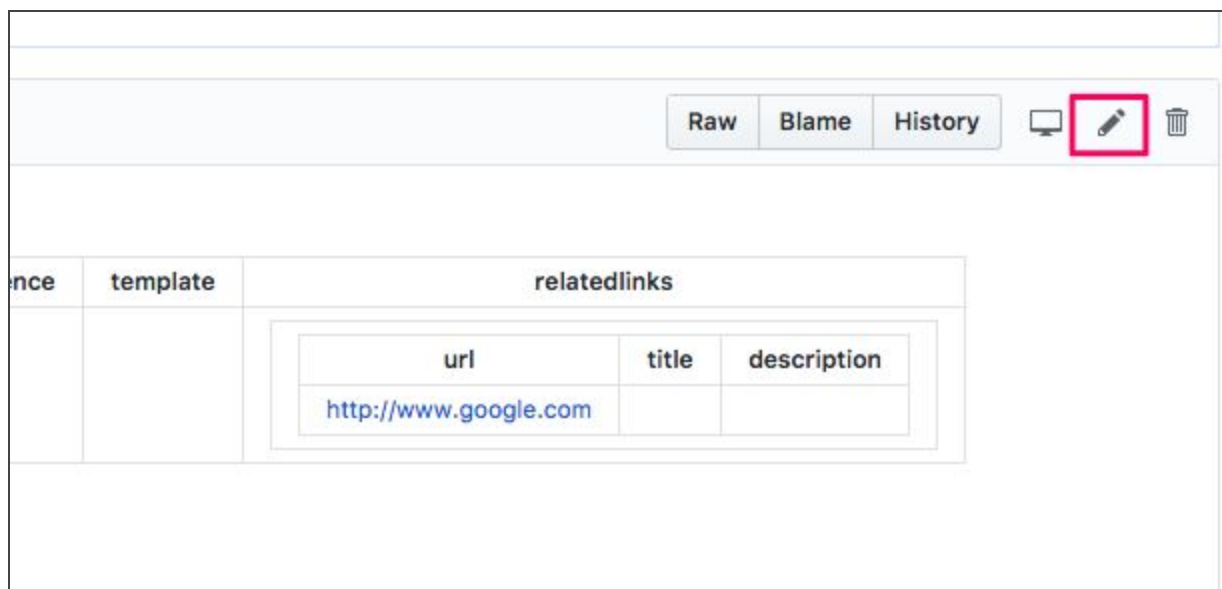📁 facilities   Facility locator

📁 **gi-bill-comparison-tool**   GIBCT (#4803)

📁 **healthcare**   Consolidated .f

📁 **housing-assistance**   updating Apply

📁 **playbook**   Update index.h

📁 profile   resolves #441

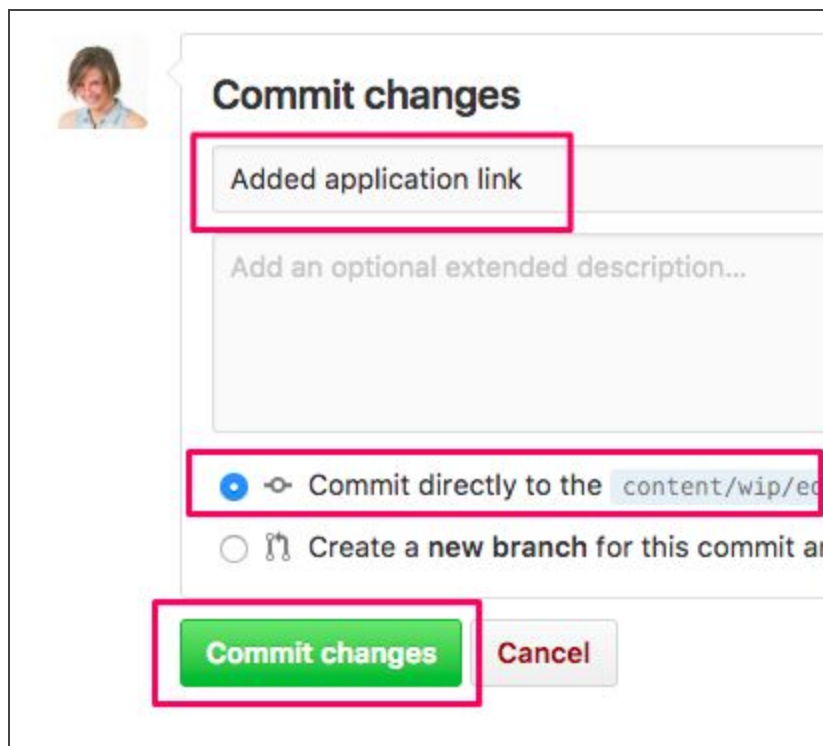👤 **rroueche** updating Apply file name

..

📁 home-loans

📄 adaptive-housing-grants.md
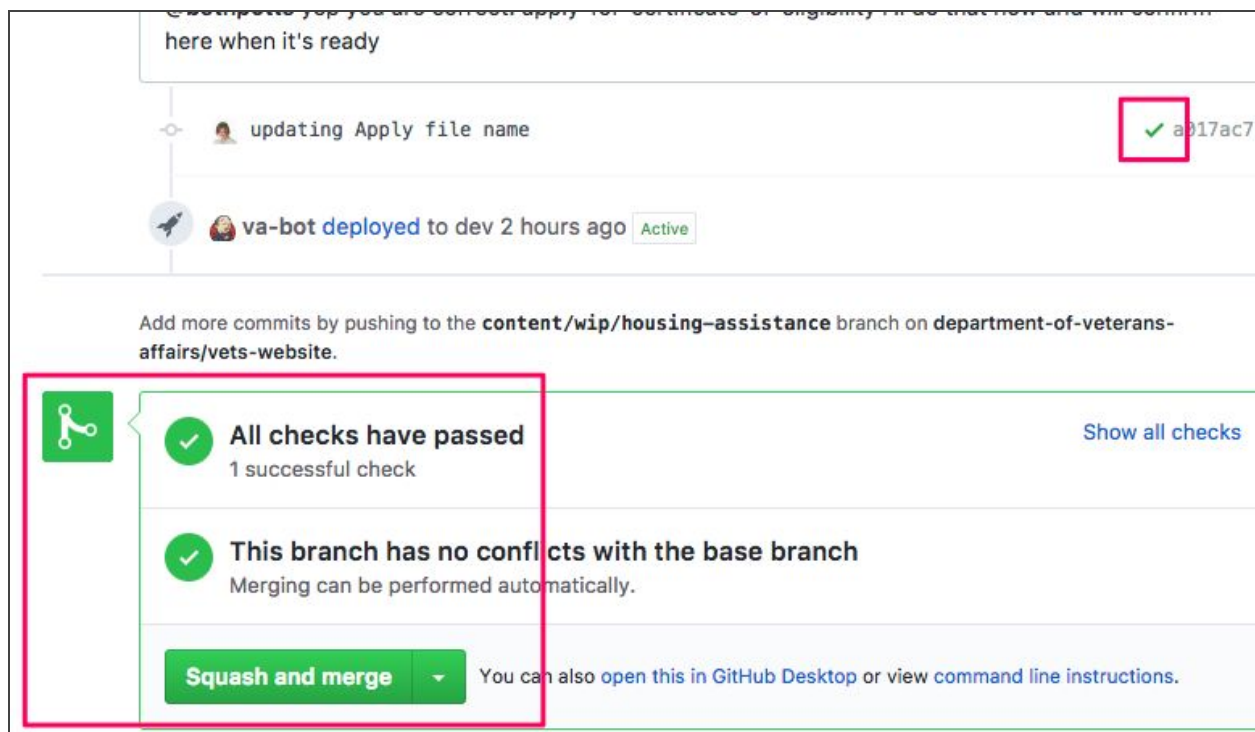
📄 home-loans.md

📄 index.md

---

4. Click the edit icon to make changes to the markdown (.md) file

5. Commit edits directly to the branch. Make sure you include a title:

6. When you make a new commit, view the corresponding PR to see if there are any errors

7. If you get a red x instead of a green check, click on the corresponding commit to find and fix the error.

8. If there are no errors, new commits will appear on the Heroku staging link about 5 minutes after they're made

Vets.gov Publishing Process

1. When you're ready for your changes to go live on Vets.gov, assign reviewers. They'll follow their own list of checks before approving and merging your PR into master. Once merged to master, the changes will be a part of the daily 3:30 PM ET deployment to the live website.