

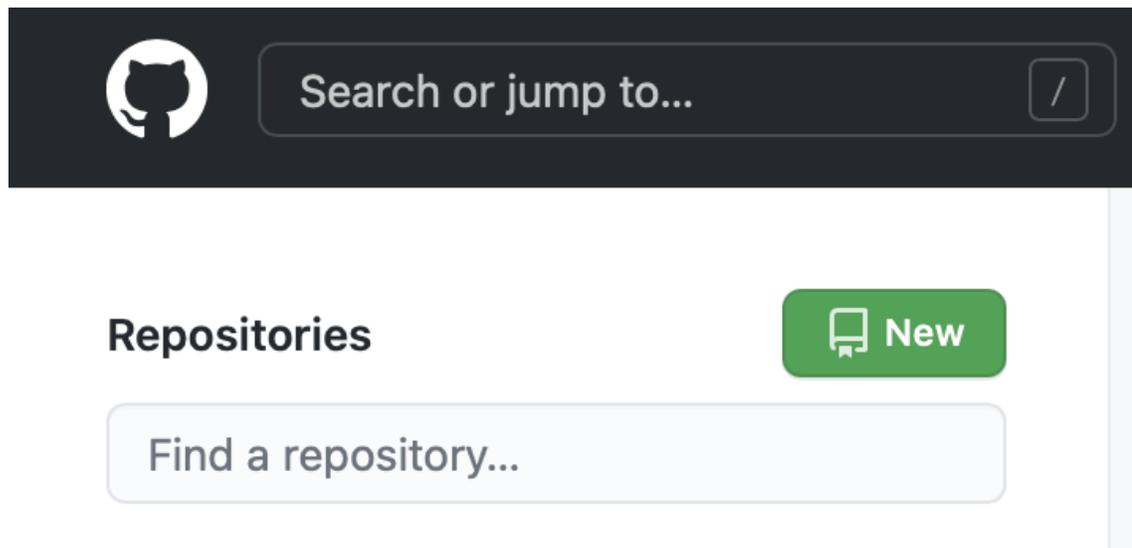
Dev 1 Git/Github Demo (2021-22)

Github Account + Software

Github Online

Getting Started

- Reminder: Github is like Google Drive for code, with some cool versioning features and strong collaboration features
- **Get started by creating a repository for your team**
 1. Navigate to github.com
 2. Click the green "new" button in the top left corner of the page



- Fill in the relevant information
 - **Repo Name** (name of the repository you'd like to create; likely your team name)
 - **Description** (Optional, but good practice for referring back to later)
 - **Public or Private** (Depends on who you want to see it; you can mark private and share with your group individually!)

- **README** (It's always good practice to have a readme file, for instructions, logs, or team communication)
 - For now, we can ignore gitignore and licensing

The screenshot shows the GitHub interface for creating a new repository. At the top, there is a navigation bar with the GitHub logo, a search bar, and links for Pulls, Issues, Marketplace, and Explore. Below the navigation bar, the main heading is 'Create a new repository'. A sub-heading explains that a repository contains all project files and provides a link to 'Import a repository'. The form fields are: 'Owner' (anupam-165), 'Repository name' (dev2-sample), and 'Description' (My sample repo for Github lecture). There are two radio button options for visibility: 'Public' (selected) and 'Private'. Under 'Initialize this repository with:', there are three checked options: 'Add a README file', 'Add .gitignore', and 'Choose a license'. At the bottom, there is a green 'Create repository' button.

- **To invite your teammates to the repo, selection:**
 1. Settings -> Manage Access -> Invite a collaborator

GitHub interface for repository **anupam-165 / dev-sample** (Private). The page shows the repository's main branch, a recent commit titled "anupam-165 Update README.md" from 4 minutes ago, and the content of the README.md file. The README content includes the title "dev-sample" and a section titled "Test". The right sidebar contains sections for "About", "Releases", and "Packages", all indicating no content is published yet. A dropdown menu is open over the "About" section, showing "Insights" and "Settings" options.

Search or jump to... Pulls Issues Marketplace Explore

anupam-165 / dev-sample Private Unwatch 1 Star 0 Fork 0

<> Code Issues Pull requests Actions Projects Wiki Security

main Go to file Add file Code About

anupam-165 Update README.md 4 minutes ago 2

README.md Update README.md 4 minutes ago

README.md

dev-sample

Test

About
No description, website, or topics provided.
Readme

Releases
No releases published
[Create a new release](#)

Packages
No packages published
[Publish your first package](#)

Insights
Settings

- Options
- Manage access**
- Security & analysis
- Branches
- Webhooks
- Notifications
- Integrations
- Deploy keys
- Autolink references
- Actions
- Secrets
- Pages

Who has access

PRIVATE REPOSITORY

Only those with access to this repository can view it.

[Manage](#)

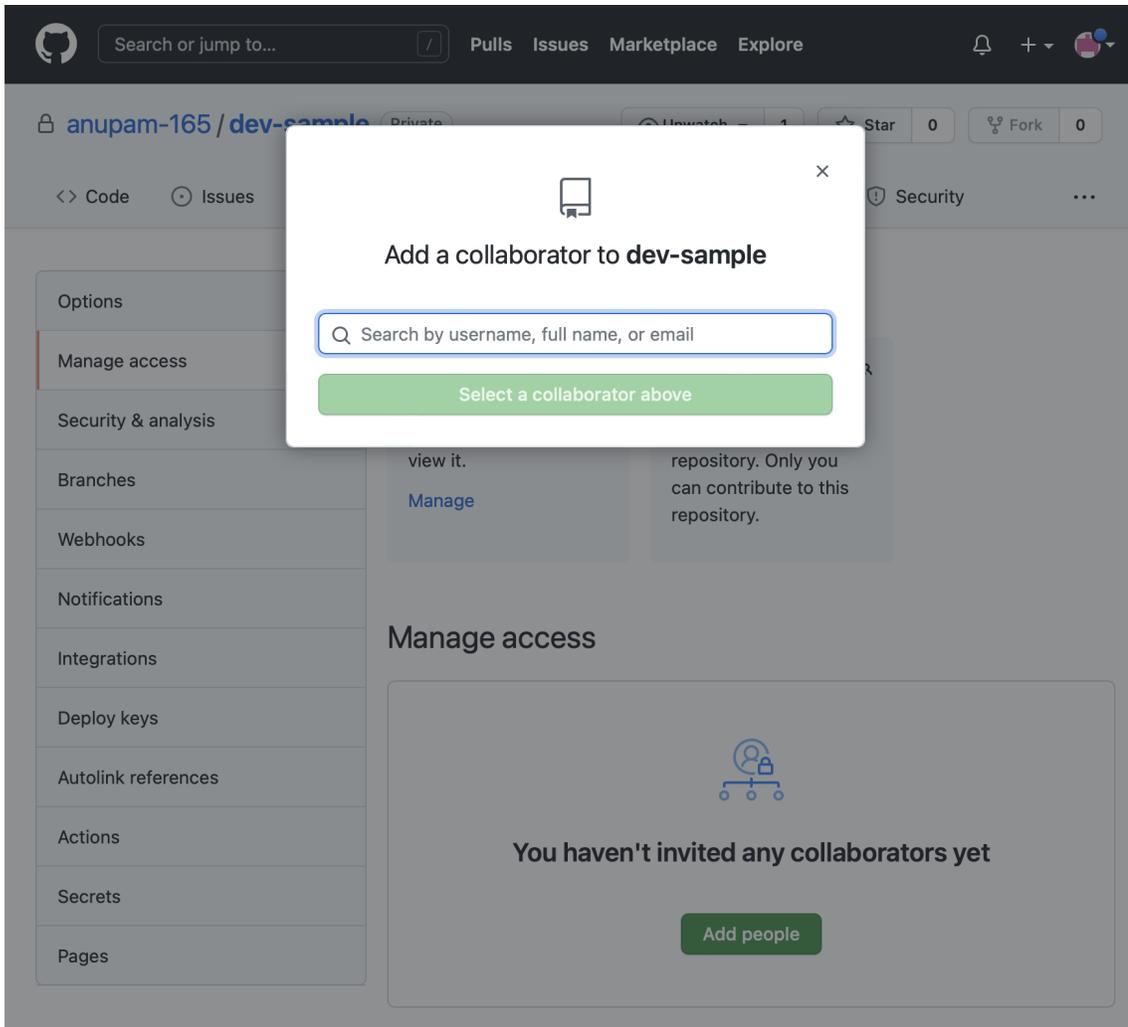
DIRECT ACCESS

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](#)

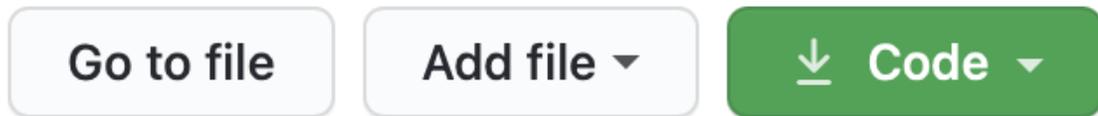


2. Confirm the invitations, and make sure that everyone accepts the invitation

Navigation

- **Readme:** Explains what's going on in this repo, and includes any important notes or details about the author
 - Can edit it like anything else using the pencil icon
 - Using a # (or multiple #) denotes something as a title or header - try out a couple combinations

- **Create a blank file in the repo by clicking "Add File" (next to the green code button), and show that it can be edited**



- **Click "Commit" to save the file**
 - Commit is generally paired with PUSH, because you commit locally, but you then push to the cloud hosted git. Since we are working on the cloud hosted git, the COMMIT takes care of this for us
 - Requirements before commit include:
 - **Name of File**
 - **File Contents**
 - **File Commit Message**

anupam-165 / dev-sample Private

Unwatch 1 Star 0 Fork 0

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

dev-sample / TestFile in main Cancel changes

Edit new file Preview Spaces 2 No wrap

```
1 Here's a sample file I created :)
```

Commit new file

My commit message here

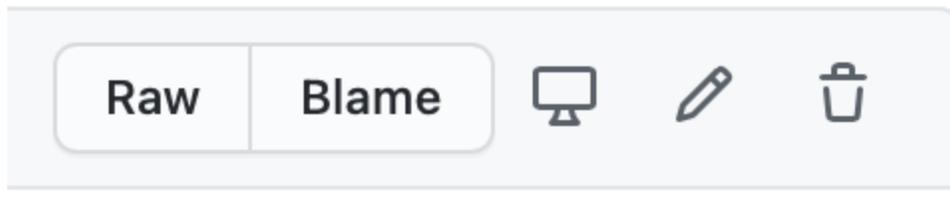
Add an optional extended description...

Commit directly to the main branch.

Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

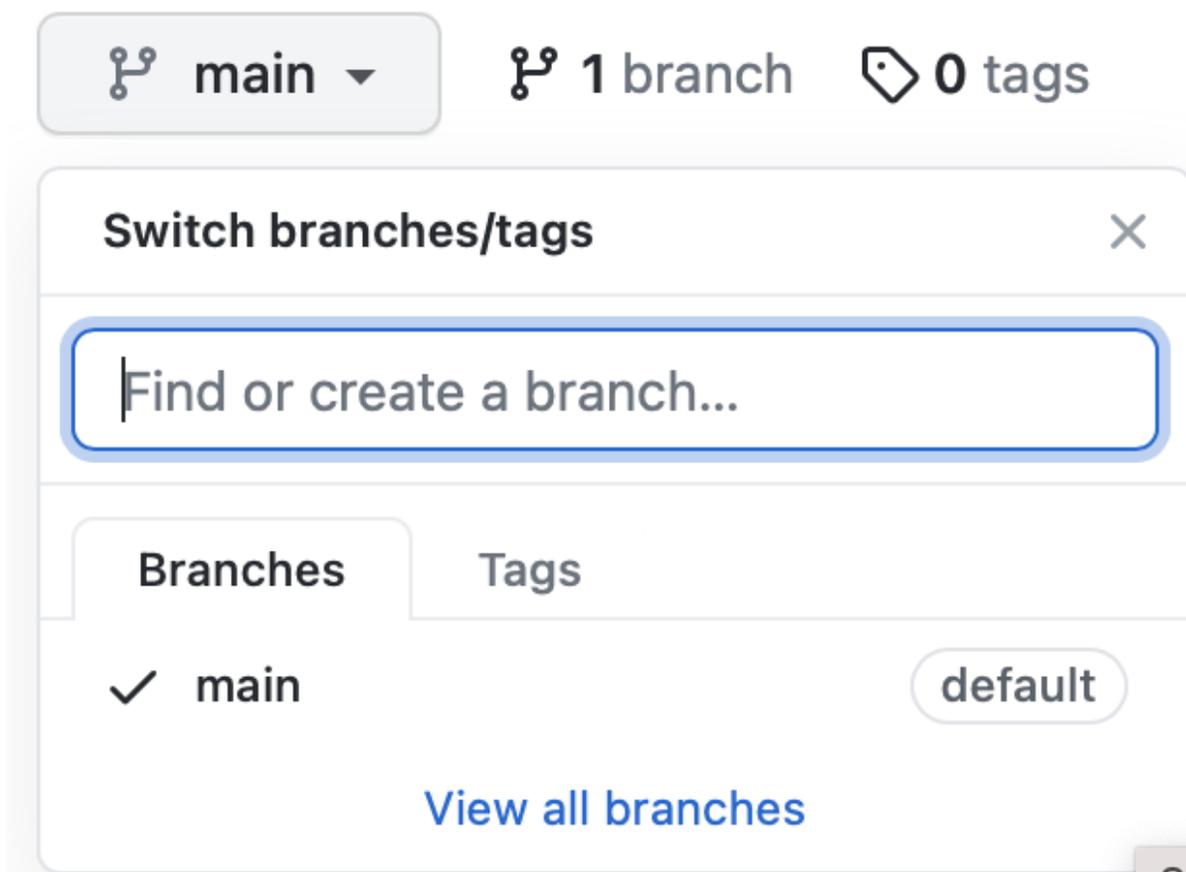
Commit new file Cancel

- To make changes to the file, you can click on the file name on the primary repo page ("Code"), and select the pencil icon in the edit menu that appears above the file preview



- **Branches**

- We said branches are offshoots of code that basically duplicate you're current main repo, and then allow you make edits and eventually merge them back
- Create a branch with a descriptive name that will duplicate your "main branch" by selecting this dropdown and typing in a new name



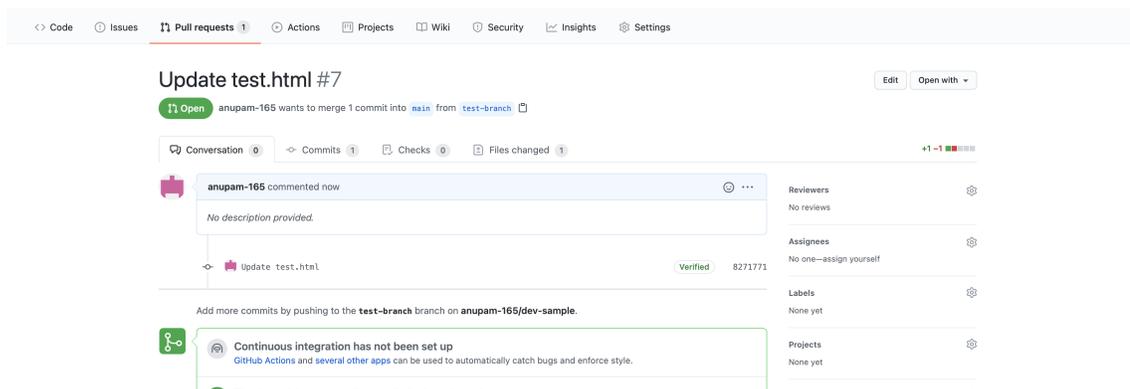
- Switch between branches using this same dropdown menu
- Changes you make in this branch WILL NOT affect the main branch
 - For example, if you create a new file here, it will NOT appear in the main branch until you use a pull request
- Once you start making edits to files, Github will remind you that you're making edits in a branch and you have to combine them with the main branch

This gives us a place to work on new things, features, bug fixes, without touching the main, working code

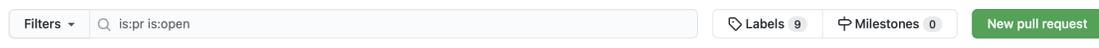
So, now that we've added a file or made edits, how do we merge them?

- **Pull Request**

- Usually, you can click "Compare and Pull Request" in the pop up that appears after editing, or you can go to the pull requests tab



- If there isn't already a pull request there, you can click "new pull request" to create one



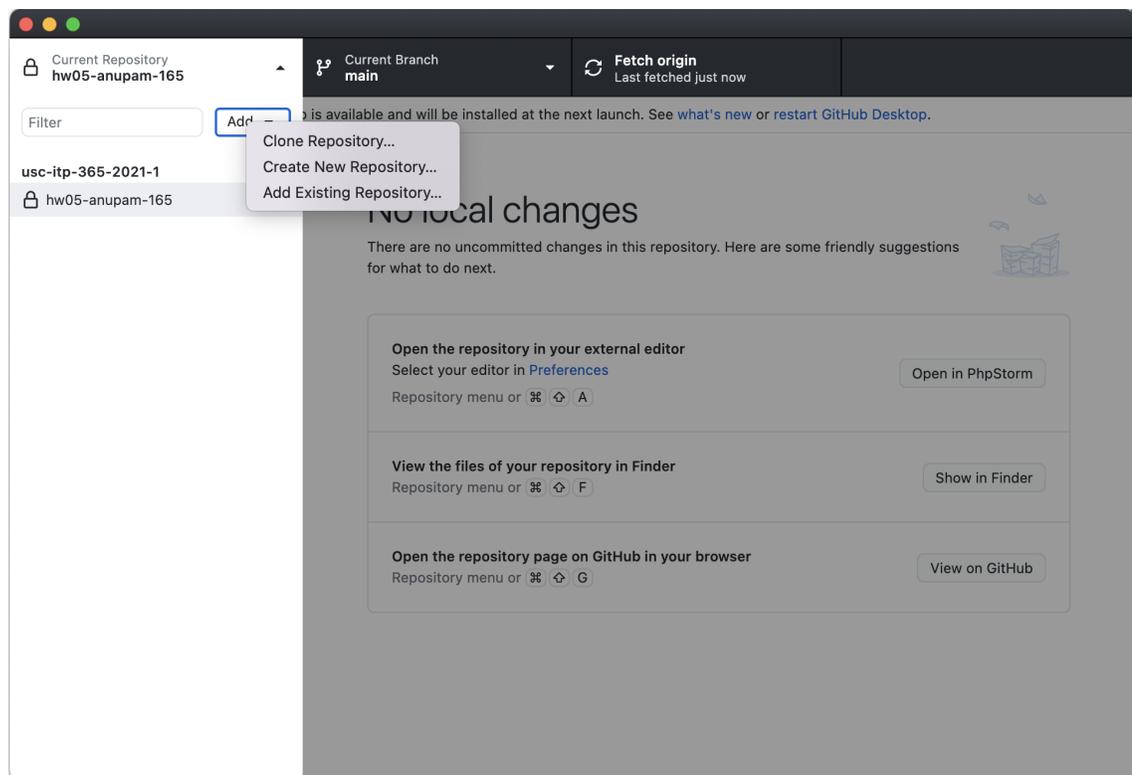
- Then, you select the branches you want to merge

- Go ahead and merge the branches by accepting the pull request (you can designate someone on your team to do this); Github will recommend that you delete the branch if you wanted to, and you could do that from here

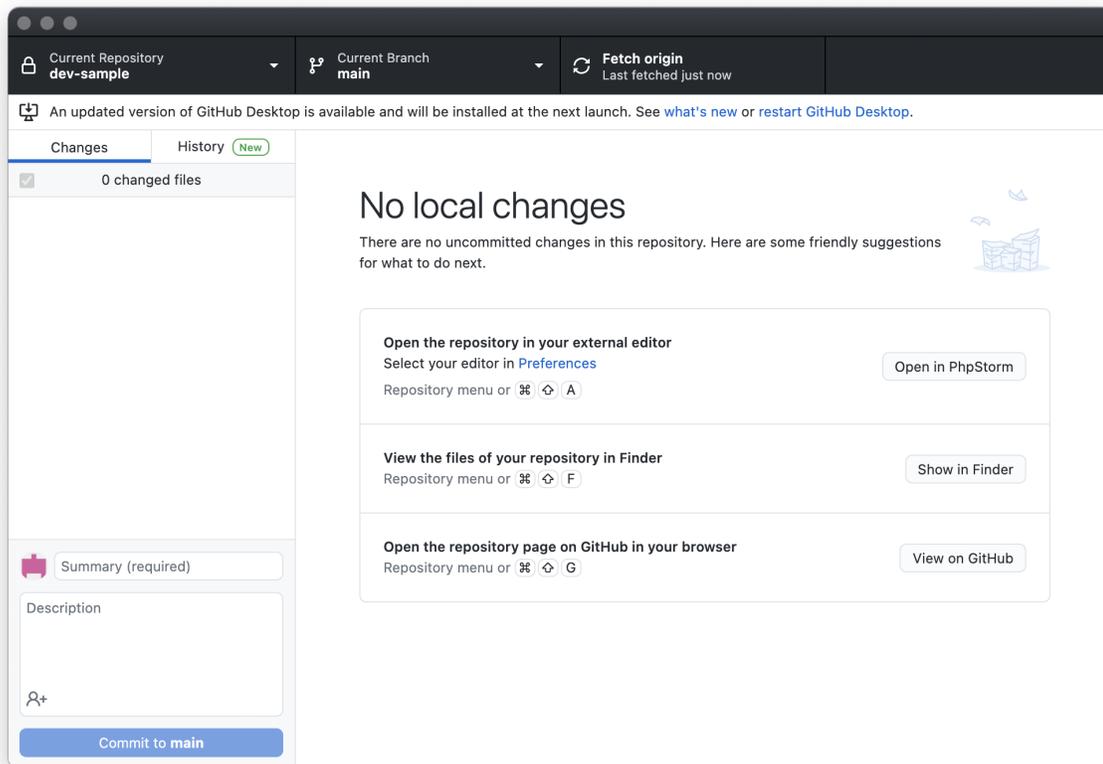
Client - Github Desktop

Github Desktop

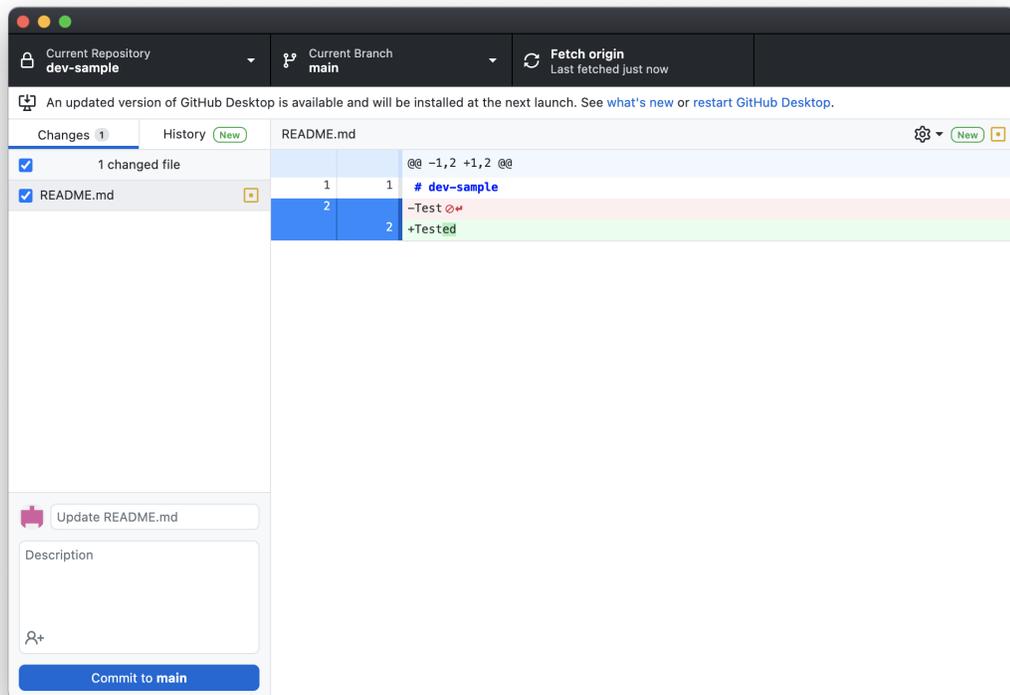
- Very similar to the Github interface, since it is a Github product
- **Duplicate your online repo to save it locally**
 - Click "Current Repository" in the top left corner, click "Add," then "Clone" and select a place to save the file



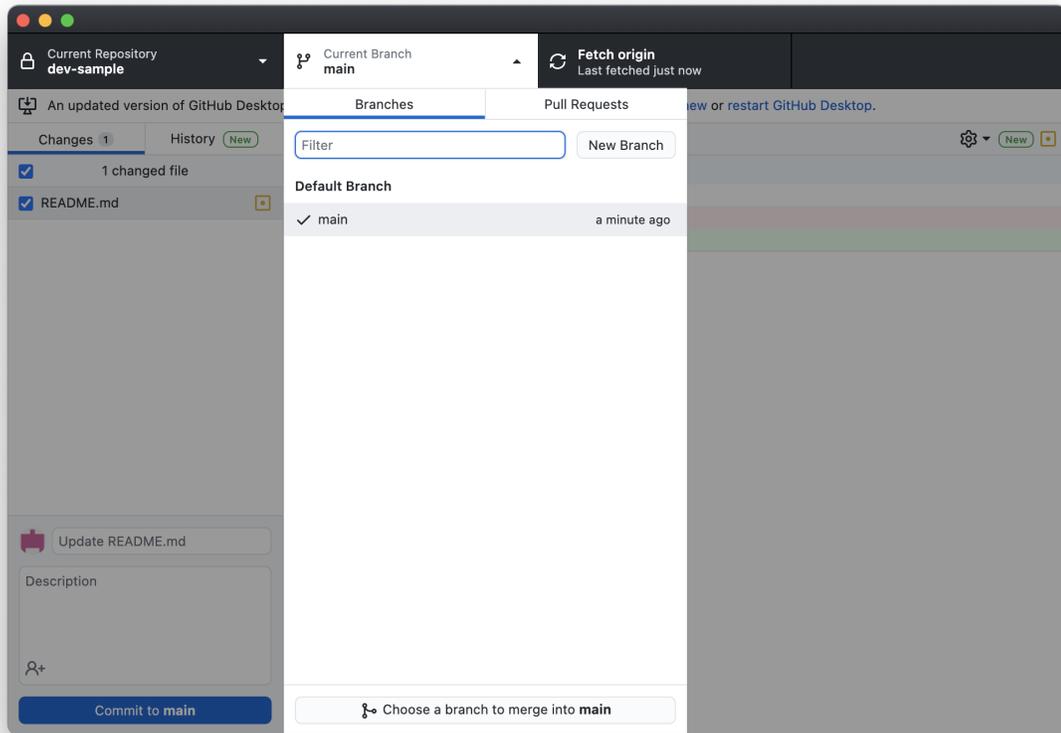
- You can make file edits, commit, and push here



- As soon as the files are changed, Github will track the changes and recommend you commit them



- You can also switch between branches using the "Current branch" dropdown



- If there are changes you need to grab from the server (you should do this before you start working everytime!), click "Fetch Origin", then "Pull"

