Generating a Pull Request

Grace Sim

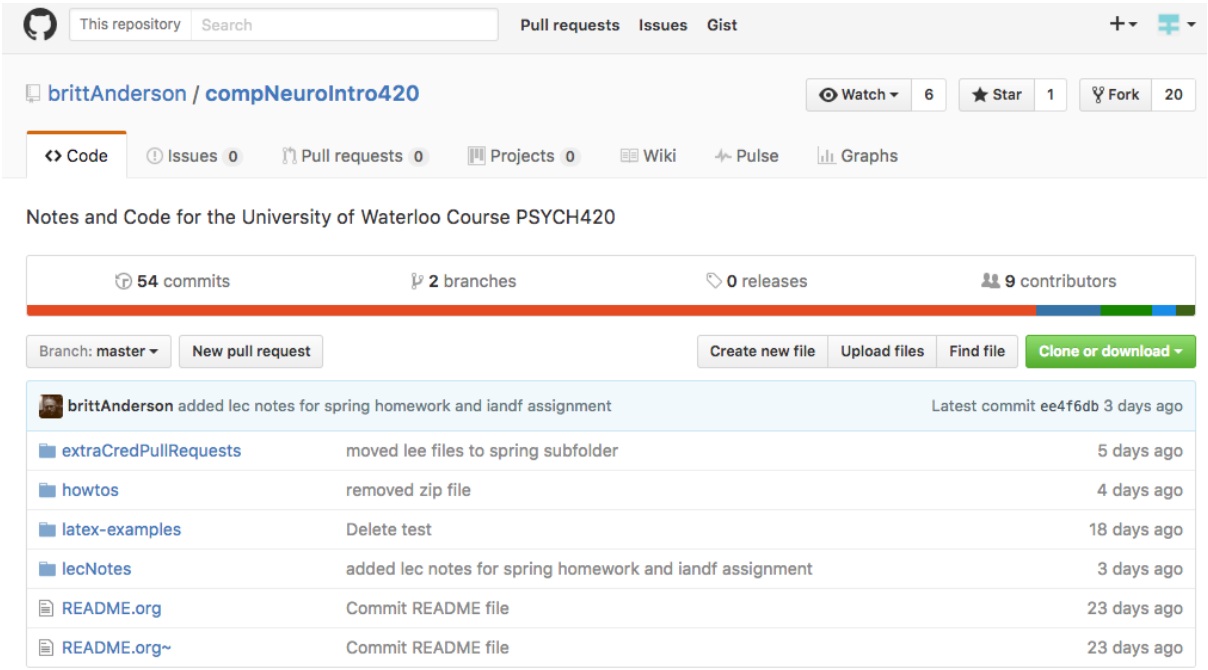
What is Git?

In short, Git is a system used to share code or documents and is useful for version control. It's like a supercharged version of Google Drive, that allows people to have their own copies of repositories (i.e., repo; essentially, folders) which they can edit without having fear of ruining someone else's work. GitHub is one of many web-based Git systems and is what we will be using.

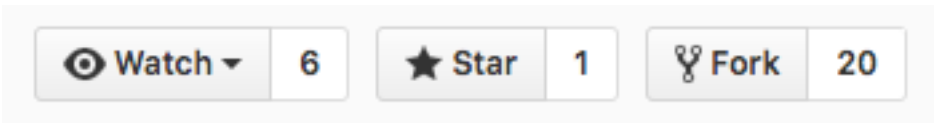
Pull Request

The following steps will help you do a pull request to a repository. This assumes you already have Git installed on your machine and already have a GitHub account.

1. **Find the repo you would like to work off of**

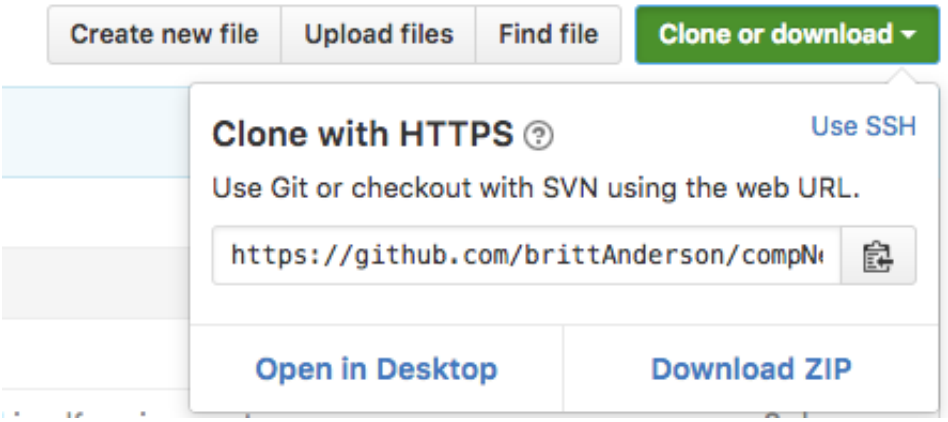


2. **Fork the repo into your own account**



This will create a copy of the repo into your GitHub *web account*

3. Clone the repo onto your machine



This will create a copy of the forked repo onto your machine so that you can easily make changes

(a) Copy the URL under "Clone with HTTPS"

(b) In your terminal enter the following code within the directory you would like the repo to reside in:

```
git clone enter-URL-here
```

4. Create a remote to the original repo so that you can track all changes from it

This allows you to track whatever changes happen in the original repo that you have forked off of so that you don't miss anything

(a) In your terminal enter the following code within your Git connected directory

Note: the remote-name can be whatever you like. Your current "remote" is the one that you have cloned onto your machine and is likely called 'origin' if you have not renamed it. So call it something other than origin. The "original-repo-URL" is the same HTTPS URL you used to clone the repository

```
git remote add remote-name original-repo-URL
```

(b) You can check that you've done this correctly by typing the following in your command line. If you see your two remotes, origin and remote-name, you've done it!

```
git remote
```

5. Make changes in your machine's Git repository

(a) Add a file that you would like to see on the master branch of the repo so that you can share your work with others

6. Add any new files to be tracked and commit those changes

If you have added a new file to the repository, you will need to add it to start tracking it, however, if you have just made changes to an existing file you do not have to re-add that file. Type the following to see the status of the tracked and untracked files in your repo:

```
git status
```

(a) Type the following into the command line to add a file to be tracked in your repository:

```
git add file-name
```

(b) Type the following into the command line to commit the changes from all the tracked files and to add a message describing what kind of changes went into the commit

```
git commit -m "Your commit message here"
```

7. Push the changes into your forked repository

This will push whatever changes you have committed on your machine to the forked repository on the web.

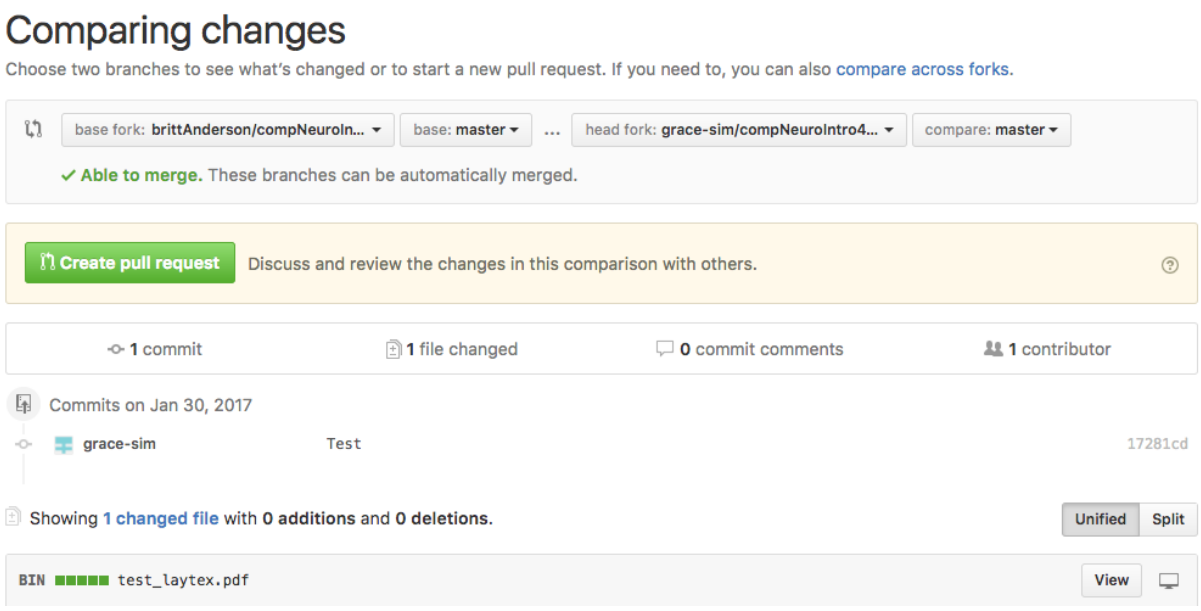
(a) Type the following into the command line:

```
git push origin
```

8. Generate a pull request

A pull request will be sent to the owner of the original repository and the changes may be merged into the new repository, if they allow it

(a) On your forked repo on the GitHub website, click "New Pull Request" and it will take you to a page that looks like this:



(b) Create the pull request

Note: If the following error appears, set up your email and username:

```
Graces-MBP:compNeuroIntro420 gracesim$ git commit test_laytex.pdf -m "Test"

*** Please tell me who you are.

Run

  git config --global user.email "you@example.com"
  git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'gracesim@Graces-MBP.(none)')
```

```
git config --global user.email "your-email@email.com"
git config --global user.name "your-git-username"
```