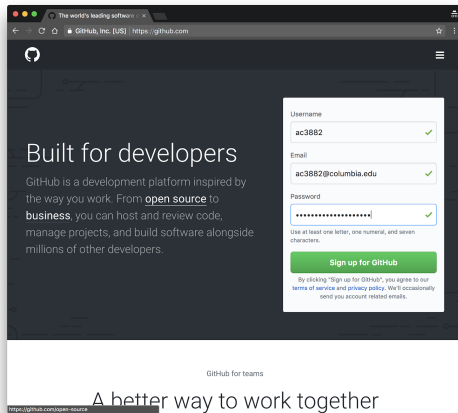


# Github Starter Guide

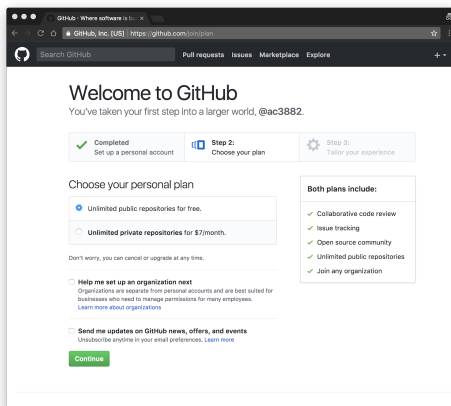
Sections taken from <http://agneschang.net/gsap-dataviz-archhum/t1.shtml>

Set up github and publish your homepage

## 1. Create your GitHub account



Go to [github.com](https://github.com) and sign up for an account. Using your UNI keeps things simple but you're free to use whatever you like.



Use default settings and click Continue.

Welcome to GitHub

You'll find endless opportunities to learn, code, and create, @ac3882.

Completed Set up a personal account Step 2: Choose your plan Step 3: Tailor your experience

How would you describe your level of programming experience?

☐ Very experienced ☐ Somewhat experienced ☒ Totally new to programming

What do you plan to use GitHub for? (check all that apply)

☐ Design ☐ Project Management ☐ Development  
☐ Research ☒ School projects ☐ Other (please specify)

Which is closest to how you would describe yourself?

☐ I'm a professional ☒ I'm a student ☐ I'm a hobbyist

Other (please specify)

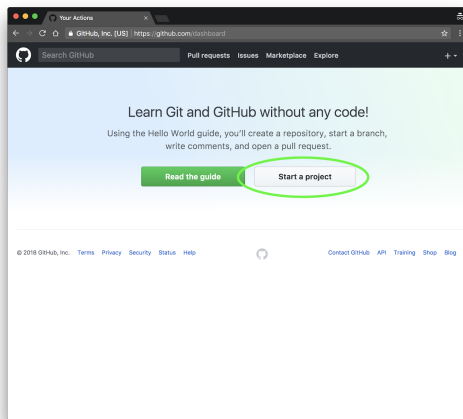
What are you interested in?

data-visualization

e.g. tutorials, android, ruby, web-development, machine-learning, open-source

Submit skip this step

Answer survey questions as you like, and click Submit.



Once on this screen, hit Start a project.

## 2. Make your class website project + install GitHub client

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner Repository name

ac3882 / ac3882.github.io

Great repository names are short and memorable. Need inspiration? How about fictional-guide.

Description (optional)

Agnes homepage for GSAPP DataViz Sp18 class

☒ Public Anyone can see this repository. You choose who can commit.

☐ Private You choose who can see and commit to this repository.

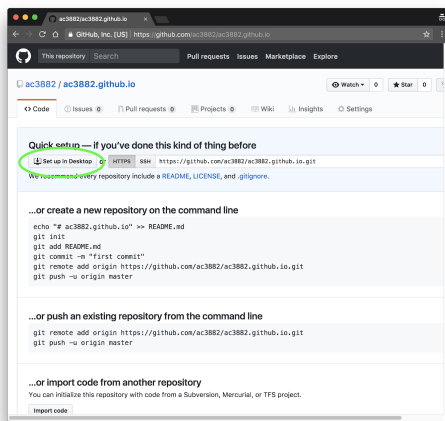
☐ Initialize this repository with a README This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add a license: None Add a license: None

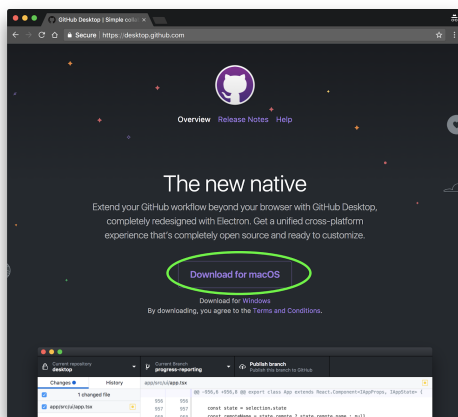
Create repository

© 2018 GitHub, Inc. Terms Privacy Security Status Help Contact GitHub API Training Shop Blog A

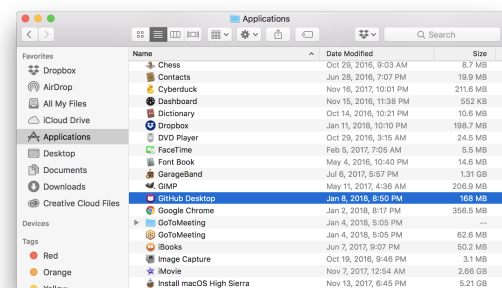
Type your repository name exactly as yourusername.github.io. (If this is not exact, you will run into problems later.) Enter a description that makes sense to you, keep the rest of the settings as in, and click Create repository.



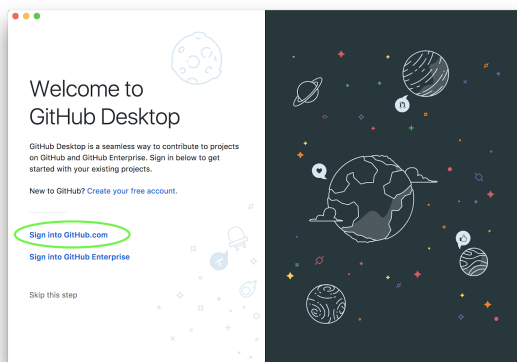
Once on this screen, click Set up in Desktop.



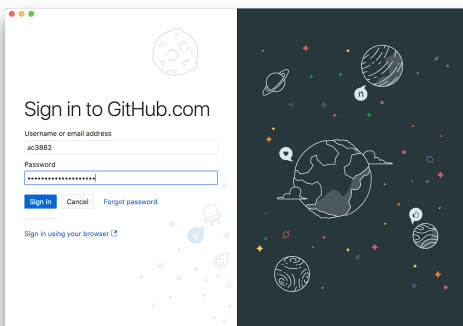
You'll be brought to this page; click on Download for macOS (or whatever OS you have.)



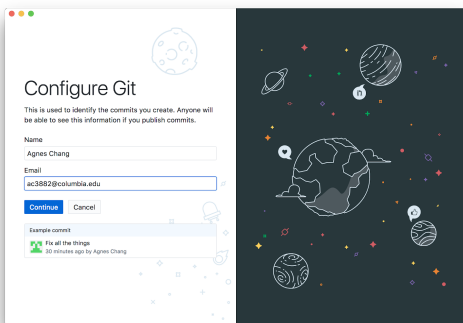
Install the GitHub Client as a normal application. If on Mac, you should have it in your Applications folder like so.



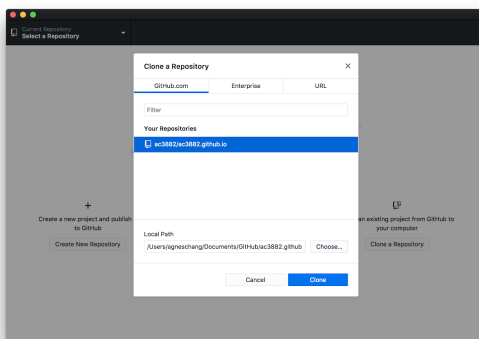
Open the GitHub client to see this screen. Click on Sign in to GitHub.com.



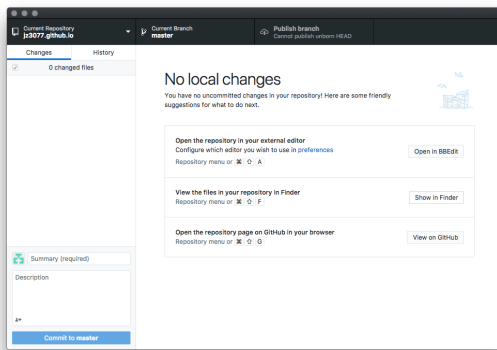
Enter your username and password.



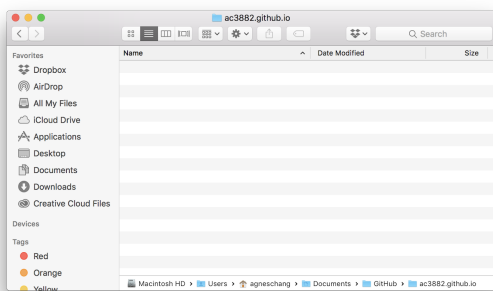
Confirm your email if prompted.



Next, you'll be prompted to "clone a repository." Select the yourusername.github.io project you made earlier, make note of where the Local Path folder is on your computer, and click on the Done button.



Your GitHub Client screen should now look like this.

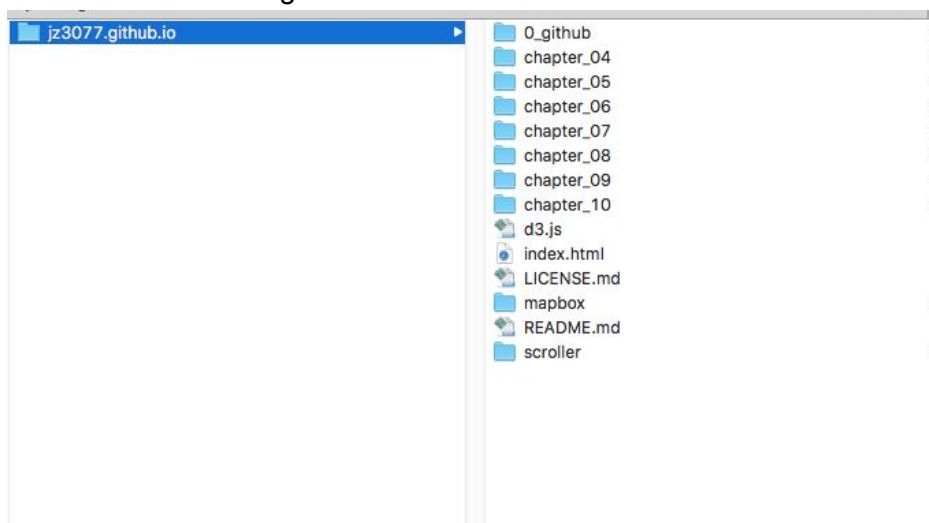


Find the folder that was in Local Path. By default on a Mac, this is  
/Documents/GitHub/username.github.io

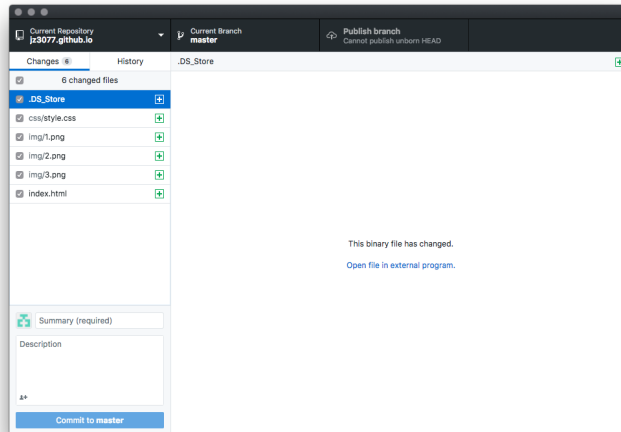
### 3. Download the tutorial files, edit and use as your site

Using /Documents/GitHub/username.github.io as your home folder.

Place the contents of your folder into /Documents/GitHub/username.github.io so that your folder looks something like this



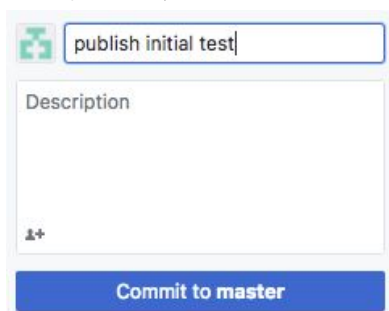
When you go back to the github client page, you will see that the files for your site: the index.html, and the chapter folders are listed on the left hand panel. \*note: the names of files may be different in this screenshot.



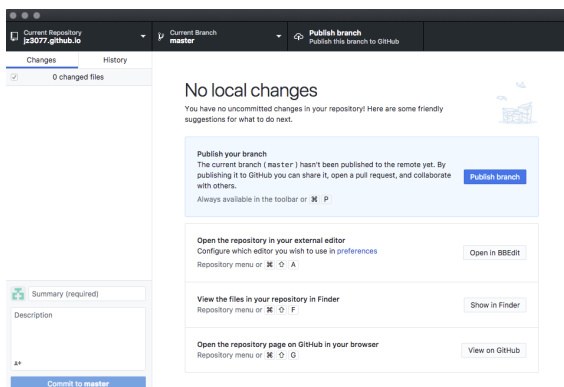
#### 4. Upload and publish your class site.

Continuing in the github client window, in the bottom left, locate the *Summary(required)* field.

Type a message in the Summary(required) field, not the Description field. The message should summarize the changes we just made(in my case, I just wrote "publish initial test", you can write what you like), and click on Commit to Master.



After you click commit to master, your screen will look like below.



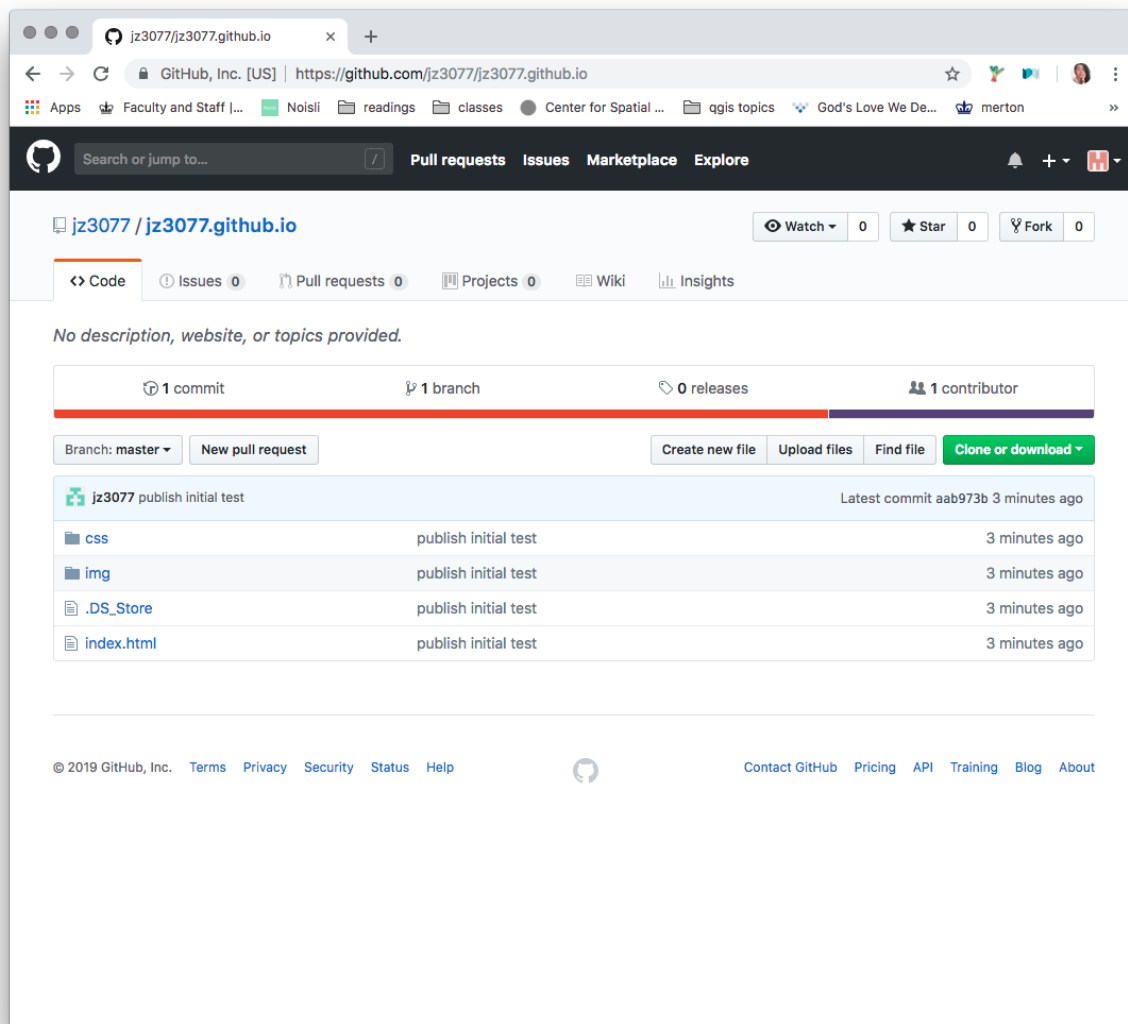
Now you can publish your branch. Click on Publish branch located in the top right. You will know that your files have been published when the publish branch button(below left) turns into the fetch origin button(below right)



5. Now open a new browser tab and go to

<https://github.com/yourusername/yourusername.github.io> and you should see this screen, listing the files you uploaded and your summary message.

\*note the files in this screenshot may be different from your actual files.



Within 5 minutes, you (or a friend) can open <https://yourusername.github.io> and voila! You've just published the site you will use for class at your own url. Submit your link to the page you published on courseworks.

*\*note do not sent a link that starts with "localhost", that is the file location on your computer.*