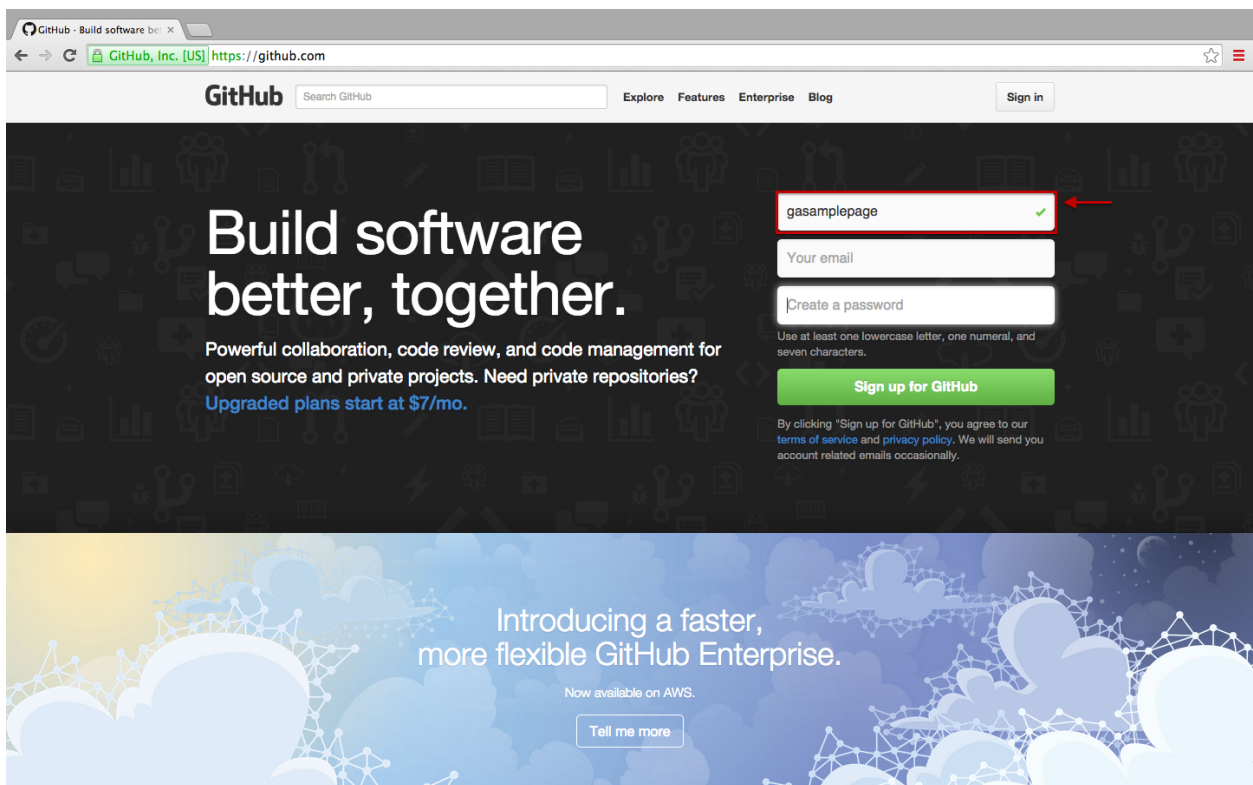## Getting Started with GitHub

GitHub enables you to host and share website files online, similar to the way Dropbox or Google Drive let you store and share other types of files. GitHub hosts the files in a remote server in the cloud and gives you a working URL so that you can share your finished site with other people.

Let's walk through this process:

Step 1: Create a GitHub account

To start, let's head over to GitHub.com. The first thing we'll do is create a username by clicking on the field at the top right of the screen.

The username you enter here will be part of the github URL for every site you host there, so choose something simple and easy to remember. Most people use their name, a pen name, or their company's name. To simplify things, try making your username one word and all lowercase.

For this guide, we're going to use the name "gasamplepage." Please pick whatever name you like, for example "johnsmith123." After selecting our username, we'll add our email and make up a password. Then we'll click "Sign up for GitHub."

Clicking that button brings us to the following page:


All we need to do on this page is make sure that "Free" is chosen as our personal plan. It should be pre-selected; however, if it's not, make sure to click on it. Then, press the "Finish Sign Up" button.

GitHub will send you an email asking you to verify your account. Take a moment to do this now. If you do not verify your email address, your page will not work.

Step 2: Download the GitHub App

Now that we've finished creating a GitHub account, we'll need to download the GitHub app. We're using a Mac, so we'll go to mac.github.com. If you're using a PC, go to windows.github.com.
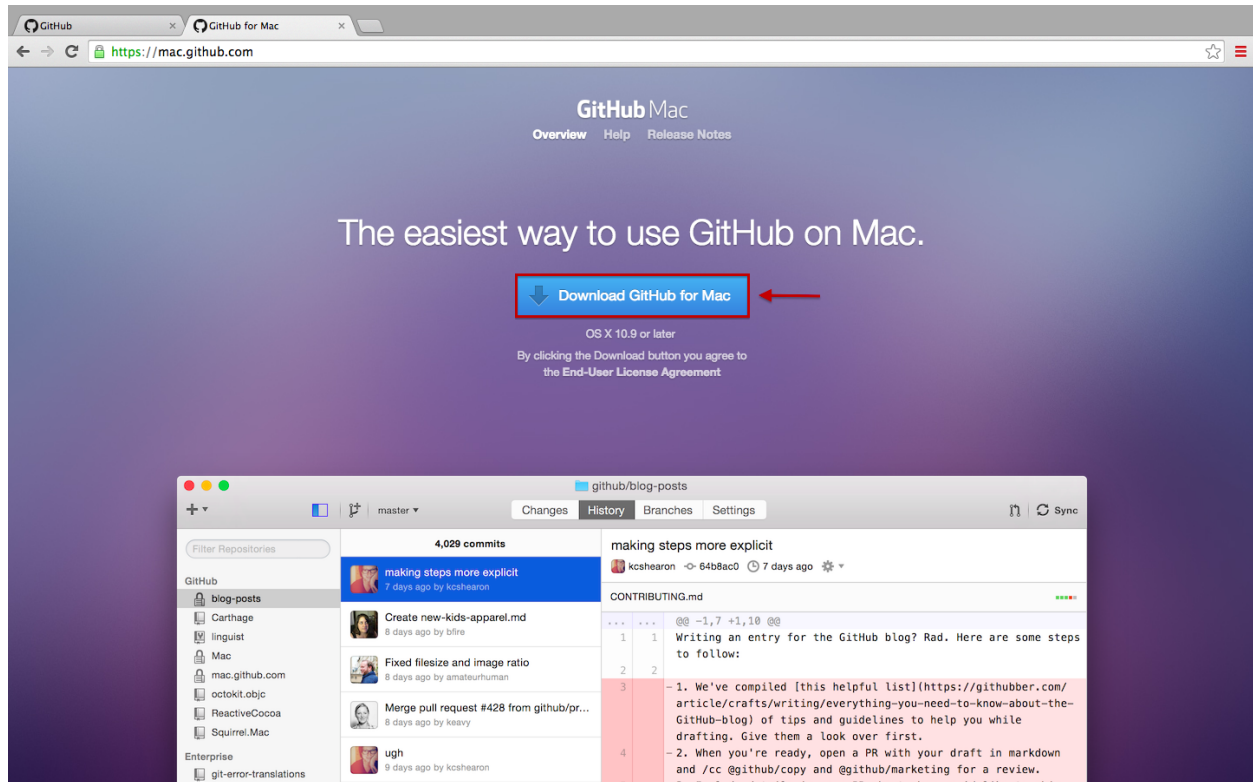
From here, click on Download GitHub. Once it's finished downloading, drag it into your Applications folder. Then, we'll open our new GitHub app.

## Step 3: Logging into the GitHub App

Let's login to the app with our new GitHub account. If this is the first time you've had GitHub on your computer, a login screen will automatically pop up. You'll log in and click "continue." Two additional pages will pop up, but you can ignore them for now. Just click "continue" again, then click "done."

If the computer you're using has had GitHub before, you'll have to manually log in. Just click on the "GitHub" tab, then click on "Preferences." Click on "Accounts," then log in.

Before we can do anything in the GitHub app, we'll first need to go back to github.com and create a new folder to house all our files.

Step 4: Creating a New Repository

In GitHub, the public folder in the cloud where you can store, update, and share all the files that make up your website is called a "repository". In order to create a "repo" (as the cool kids say), click on this green button that says "New Repository."
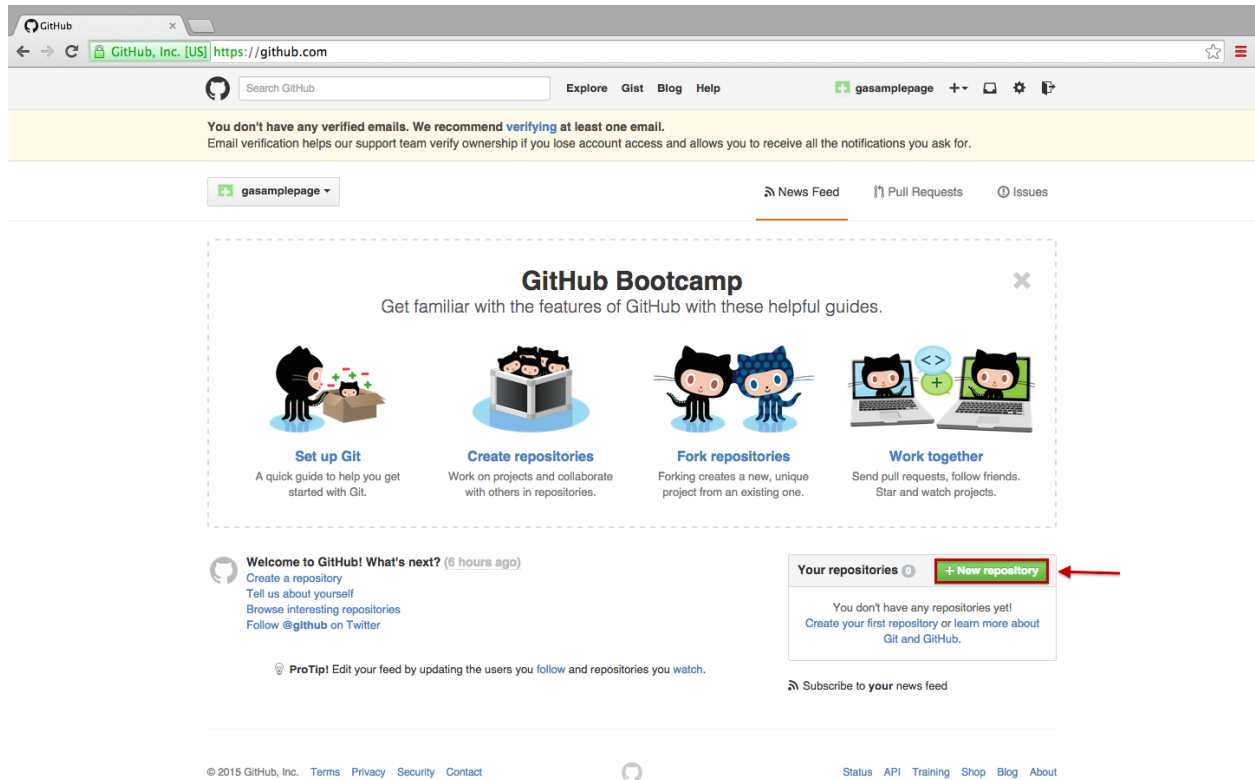
Step 5: Creating a New Repository part 2

Clicking that button should bring you to the new repository creation screen, which should look something like the following image.

The first step we'll take on this page is to name our new repository. Feel free to choose whatever name you like. Just know that if your name includes spaces, GitHub will automatically convert them into hyphens for you, since spaces aren't allowed.

To keep it really simple for this example, we'll just name our first repository after our user name: "gasamplepage."

After naming our new repository, we don't need to adjust any other settings, so we'll just click on "Create Repository."

Step 6: Setting up in Desktop

Yay! We made a new repository. Our next big step is to add our website's files to it. To get started, click on the "Set up in Desktop" button. This brings us back to the GitHub App we downloaded earlier.

Step 7: "Cloning"our Repository

When you re-open the app, a message should pop-up that says "Clone As" with the name of our new repository pre-entered. Basically, "cloning" means that we're copying our GitHub repository from the cloud and saving it as a local folder on our computer.

Doing this will allow you to add and edit files locally, then sync up those changes to your GitHub repository in the cloud when you're done.

Save this new repository folder somewhere that will be easy to find, like your Documents folder or your Desktop. Then, click "Clone."

This will produce a local folder with the name written in the "Clone As" field, and it will be located wherever you decided in the "Where" field.
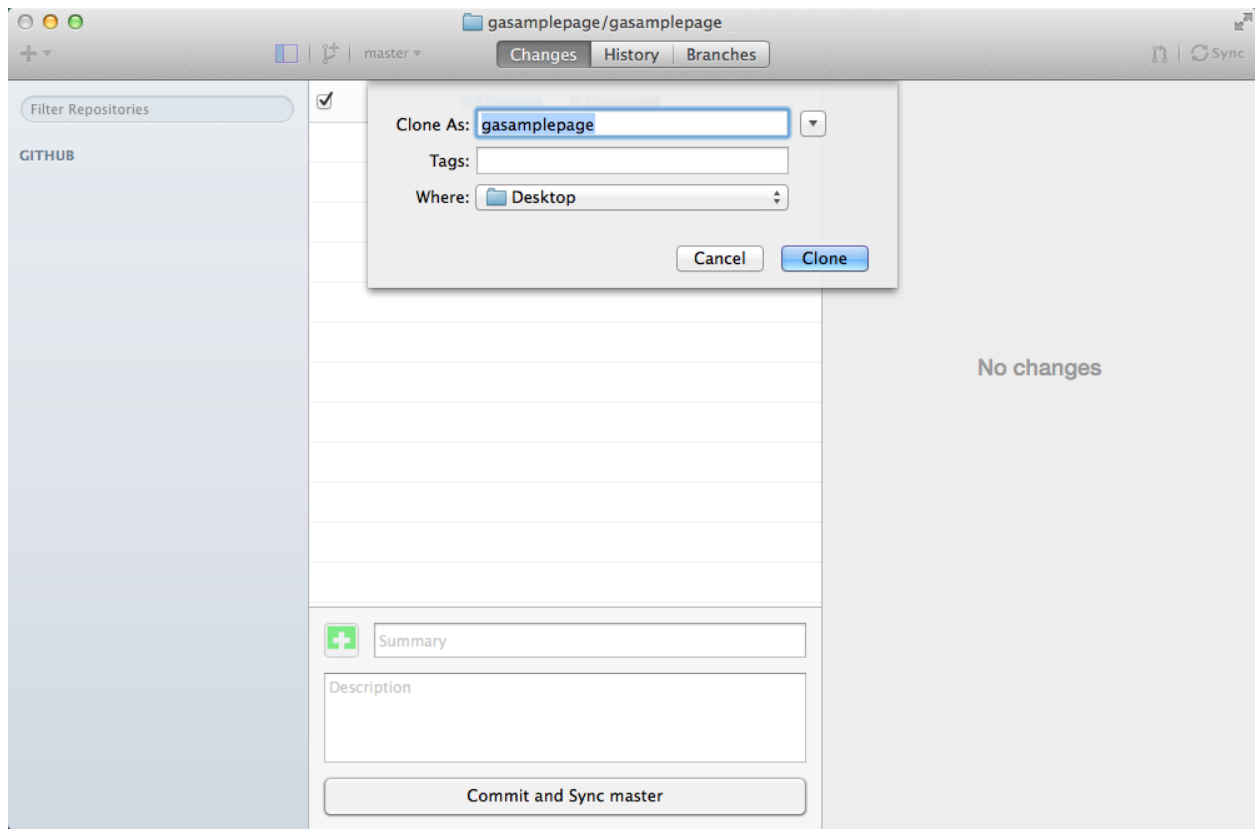
Step 8: Adding Files to our Repository

Shrink the GitHub app and find the cloned local folder you just created. If you saved it to your desktop, then go to your desktop. If you saved it to your documents folder, go there. etc.
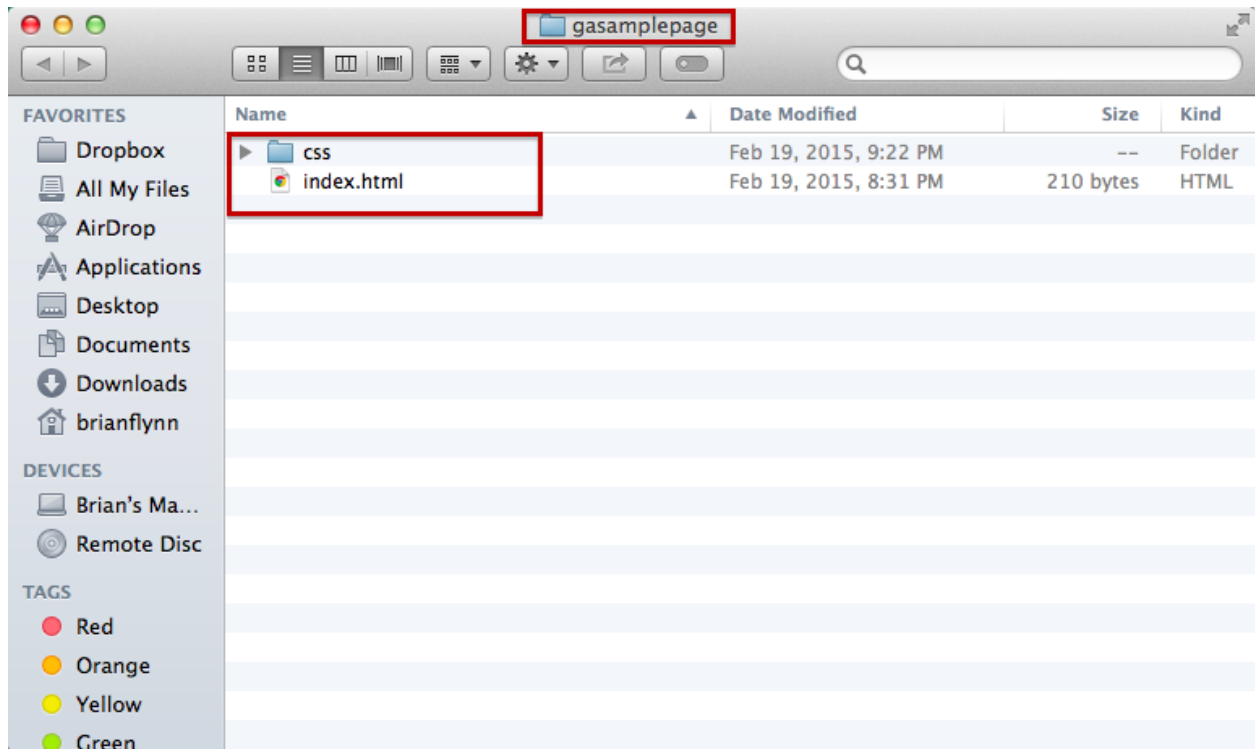
Once you find your new cloned folder, put the HTML and CSS files making up your web page inside of it, preserving the directory structure.

Once you do that, go back to the GitHub App.

Step 9: Syncing our Repository

Depending on your software, the layout of your GitHub app might look different than what's seen below. Don't worry, for our needs, the difference is purely cosmetic, and the steps are essentially the same. Just take the following steps, and you'll have your page online in no time!

(1) Make sure you're in the section labelled "Changes."

(2) In the summary field, type a message describing this update. We'll write "syncing my new repository!" Later, when we're making updates, our summaries might look like "changed the background color and logo size per client's request."

(3) Press the "Commit to Master" button.

**(4)** Click on the "Publish" button.
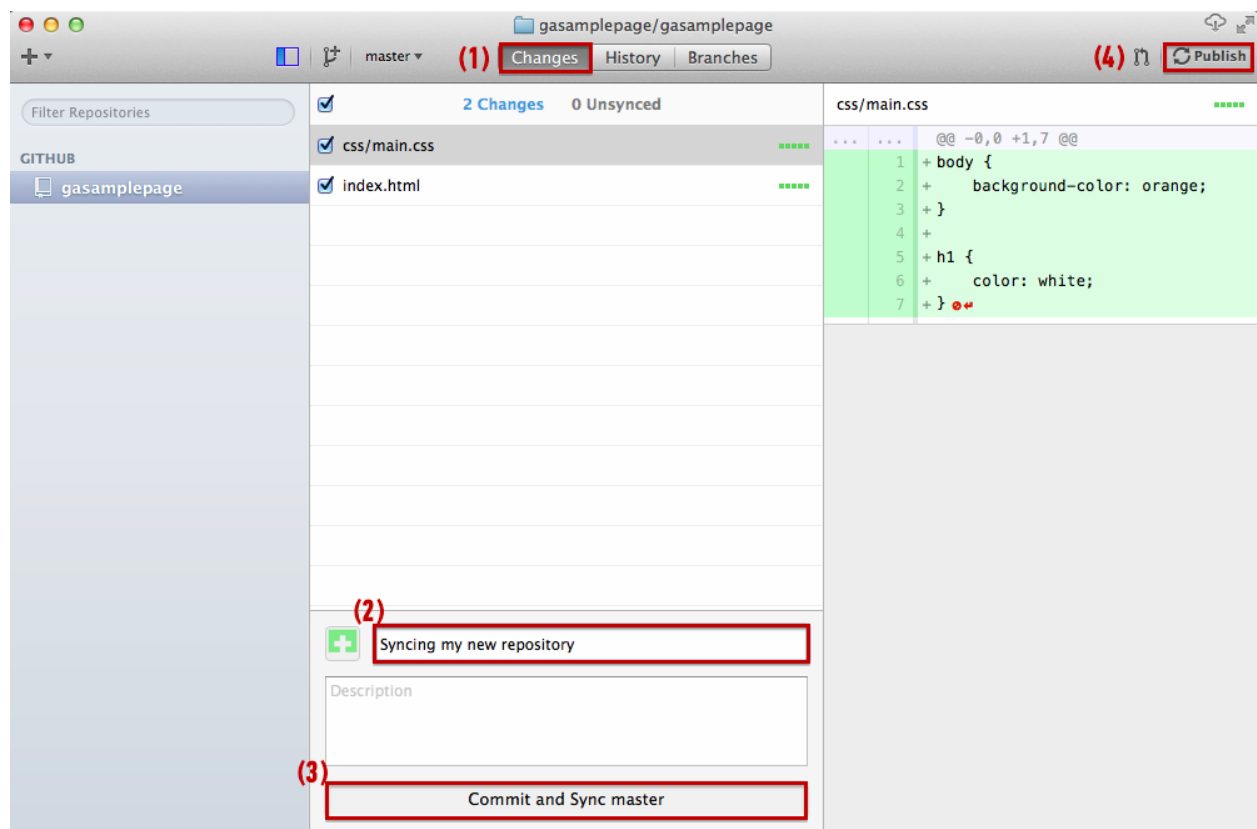
Step 10: Loading Your Page (aka OMG We're Finally Done!)

Unfortunately, the very first time you commit and sync a new repository, it can take anywhere between 0 and 30 minutes for GitHub to get your files up online. The good news is that every time you sync after this, it'll be pretty much instantaneous.

Once that's done, the final step is to visit our newly hosted files, so let's go back to our web browser. In the URL bar, we'll type in github.com, followed by a slash and then your username. So in this case, I'm going to type in github.com/gasamplepage.

What you see now is your GitHub profile page. In the center of the screen, we'll see the repository we just created. Let's click on that.
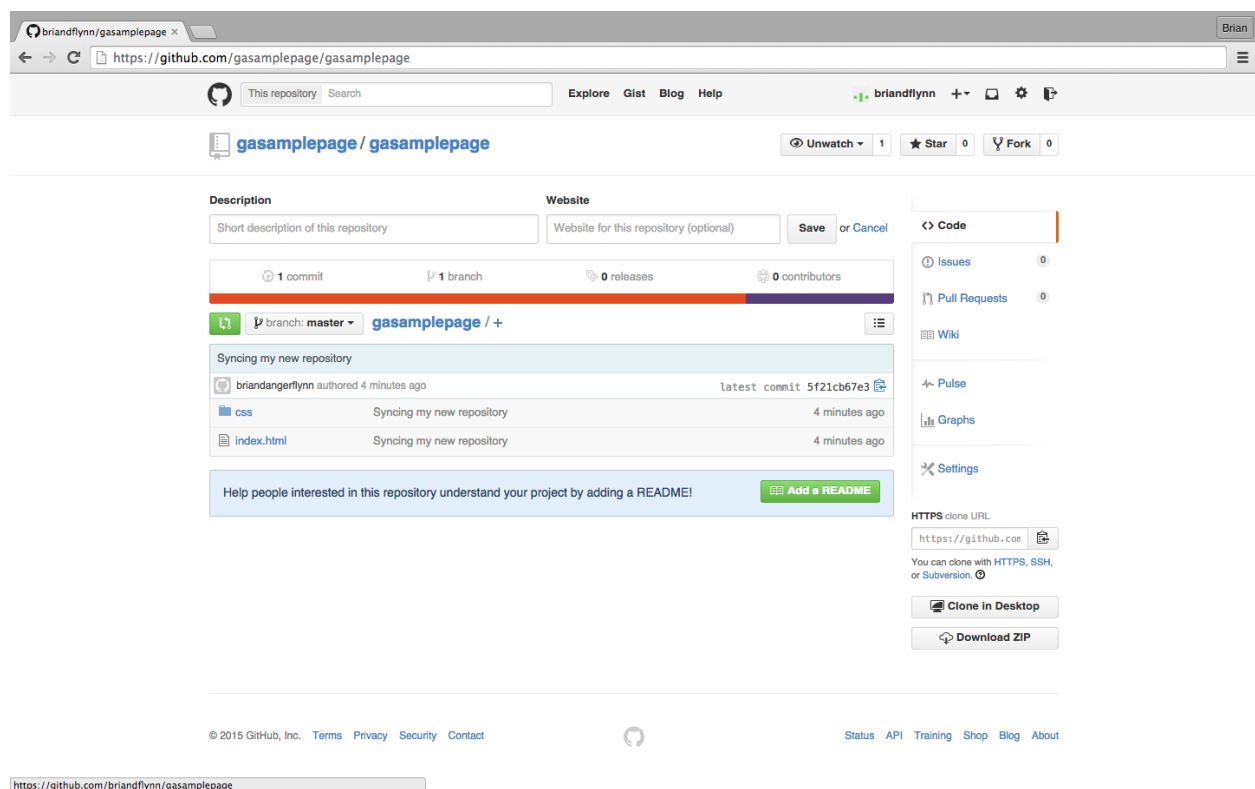
Great, so here we can see our new repository with all of our files in it. If you like, you can click on any of these individual files and GitHub will show you what's inside. We can share a link to this page with anyone we might want to share our code with, for example, your instructor, TA or mentor.

If you ever want to update or adjust the files in your repository, just change the files as needed, and then re-commit and sync them in the Github app.

THE END