Github and GitLab Overview

Alejandro Martinez

East Los Angeles College

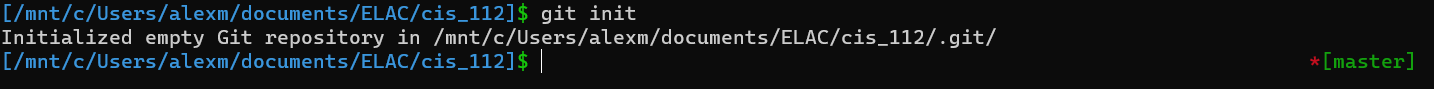
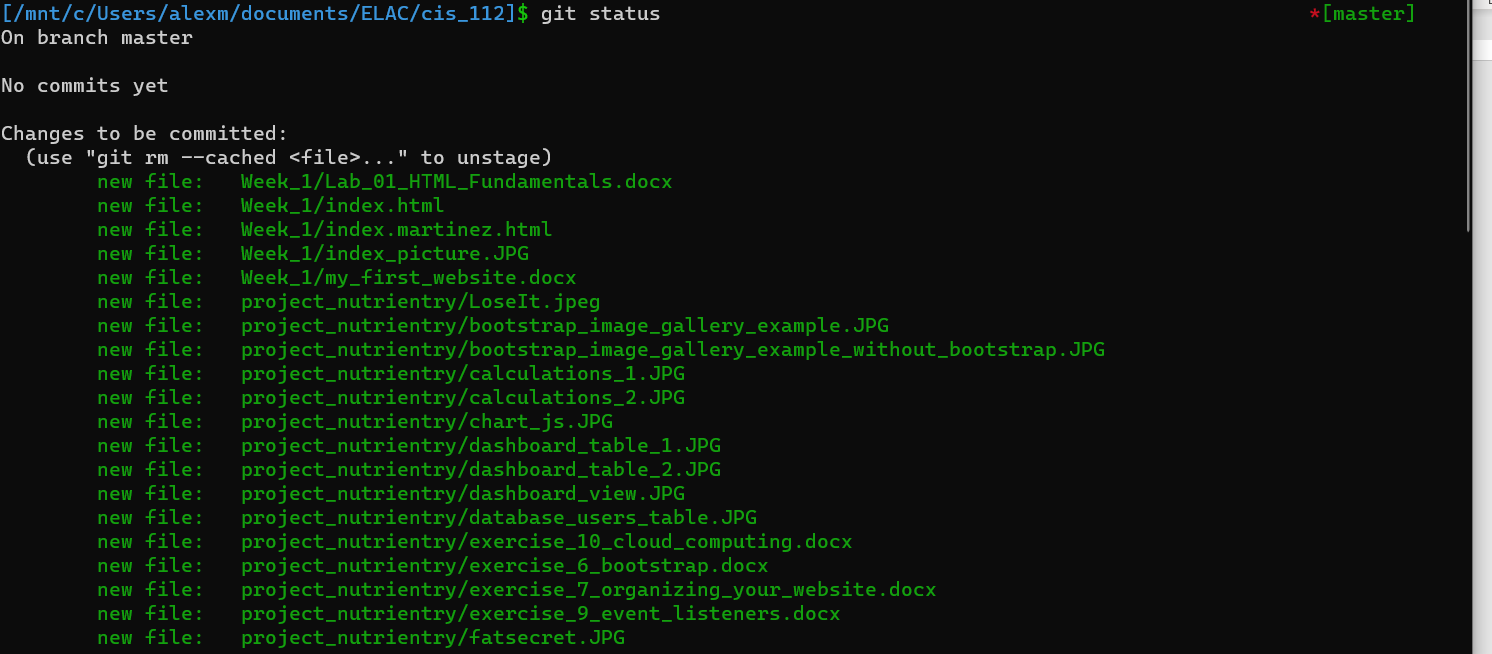
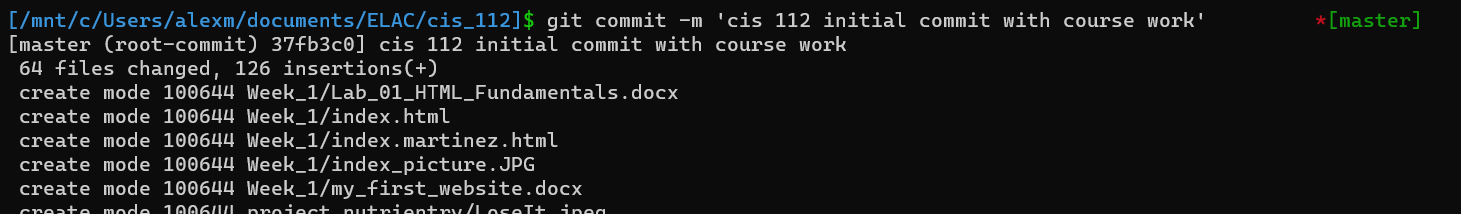
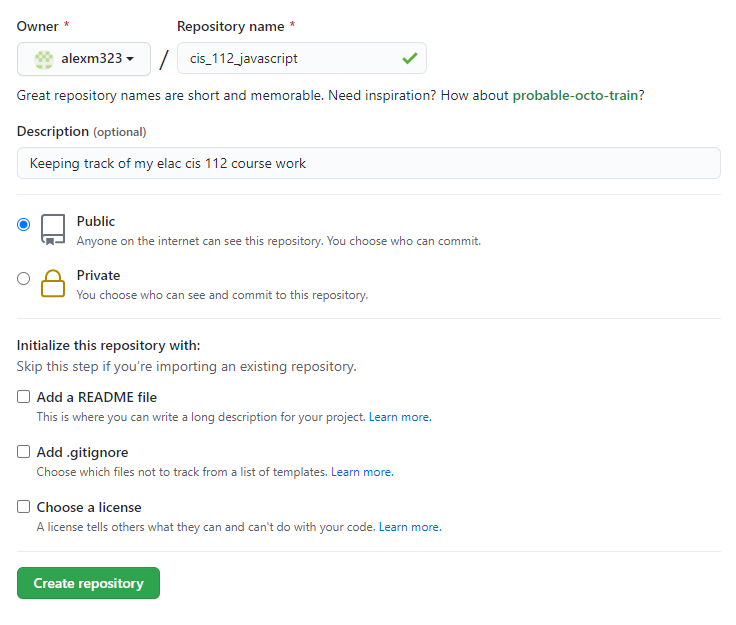
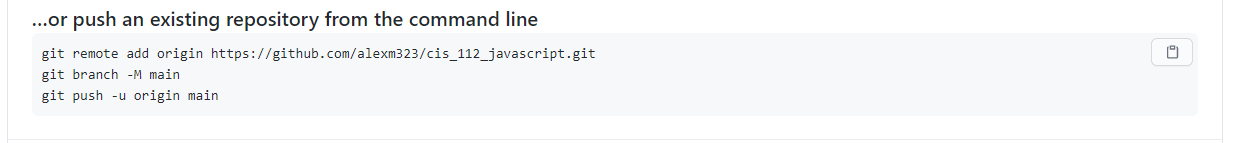
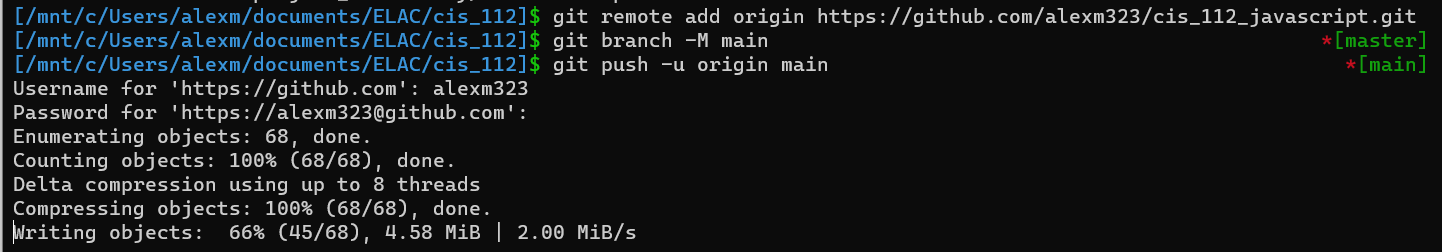
CIS 112: JavaScript Programming

Professor Rodriguez

Introduction

To understand Github and GitLab, we first need to understand what a version control system is. Specifically, we want to know what Git is. Git is a version control system, which is a tool used to save and modify version of an existing code program. You are able to track version changes according to branches so if you have a specific change that you want to try but you don’t want to break your code, you can create a branch where you can make changes to the code without compromising the working master/main branch where the working code lives. First let’s talk about Github

Github is a place where repositories can be made to keep track of projects and have people collaborate and contribute to existing projects. So the word Git in Github is referring to the version control system that is being used, and the hub portion of the name represents the idea of people coming together to complete projects. This is the idea behind Open Source software which is software that has all their code out in the open and people can look at and contribute to different projects that they are interested in. You can think of Github as something akin to Google Drive, where you are able to upload and access your files anywhere. By uploading your files to github you can easily download and use them on any system in case your machine crashes or becomes corrupted. Below I will detail how to initiate a repository on your machine assuming that you have Git already installed globally.

1. On the command line , you are going to want to maneuver to the folder on your system where you are going to want to track your project. For example, if you wanted to create a folder to keep track of all of your school projects for this class you might have a folder called CIS\_112.
2. Once you are in the folder where you want to create your repository, you are going to want to use the command *git init*. This will initiate a repository in the current folder and it will tell you what branch you are currently on, which should be master.
3. Once that is complete you can add any files to the staging area where the files will wait to be committed which is when your files are actually saved. Any files added and committed to the staging area will be tracked by git and that allows you to see different changes between versions. For this we use a command called *git add* . Git status is a command that allows us to see the current pending items in your git repo.
4. To finalize the files that you want to keep track of you can use a command called *git commit -m “Insert your message here”*. The -m flag allows you to place a comment on your commit so that you can later look back and this also allows other users to see what changes were made on this file.
5. Now that we have our master branch we can move on to putting these files on Github. You will need to make a github account and once that’s done you can start a new github repo which will be connected to your git repo in your local machine.
6. Once you’ve done that the next screen will show you the final commands you will need to input to get the files up on github. 
7. Once you run it in the command line you will see git and github connecting to upload all of your files. 
8. Once that’s done you can now access your files on github!

Github is great for collaboration but you will often times need other tools for reviewing code, making changes, giving users access, etc. This is where GitLab comes in. GitLab is a one stop shop for your dev ops needs. Dev Ops is a combination of things that is summarized by an organization’s ability to deploy applications quickly and securely. GitLab intends to solve the problem of having multiple platforms for different stages of the development process and aims to be an all inclusive solutions for businesses that want to ramp up their development time by cutting down on the need to switch between platforms to give users access, test security features, merge version controls. GitLab looks like an awesome tool but is a little beyond the scope of what I am currently capable of, but it is very interesting nonetheless.

Cloud computing is a way for businesses to avoid the hassle and complexity of setting up their own servers and to power their applications based on their specific business needs. One of the biggest proprietors of this kind of service is AWS , amazon web services. This shift for Infrastructure as a service has also lead to the advent of software as a service. This is why you often see the decline of products that you can actually purchase and now need a subscription for almost any software service. Businesses no longer have their own servers and instead rent hardware space on the cloud which is just a distributed network of computers working in tandem to provide uptime for a company. I am personally not a fan of no longer being able to own software like adobe photoshop and now needing to pay a monthly subscription fee in order to use it. There are also issues with the providers of cloud computing since to be able to compete you basically can’t avoid using a big tech company’s services at some stage of production .

# References

Git Lab. (2020). About Git Lab. *https://about.gitlab.com/*.

Novoseltseva, E. (2018). Benefits of Using GitHub. *DZone*, 1.

Steve Ranger. (2018). What is Cloud Computing? *ZD Net*, 1.