Github and GitLab Overview

Alejandro Martinez

East Los Angeles College

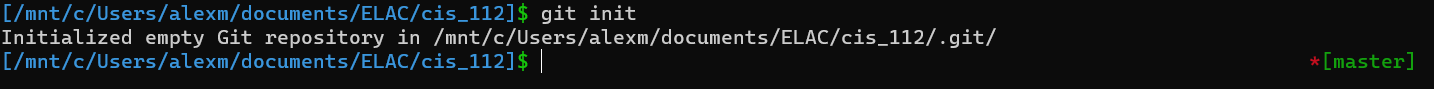
CIS 112: JavaScript Programming

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* .