Steps to use Git Hub desktop app, VScode and to have your work recognized as your own on the website github.

Programs required to follow these instructions

1. Visual Studio Code
   - https://code.visualstudio.com/download

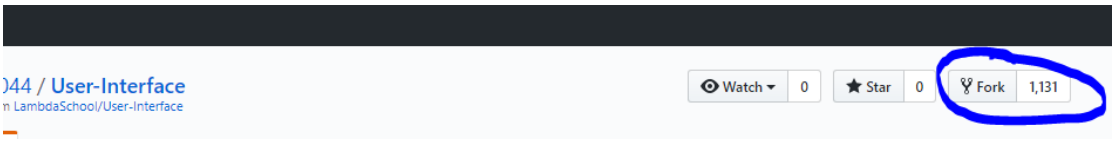
2. Git Hub Desktop App
   - https://central.github.com/deployments/desktop/desktop/latest/win32

3 Visual Studio Code Add-ons

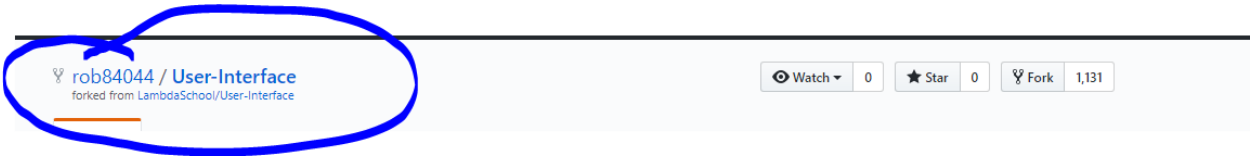
1. Bracket colorizer
2. Live Server
3. VS live share
4. Bracket Pair Colorizer

These instructions assume that you already have those programs/extensions installed. Please install them, then continue on.

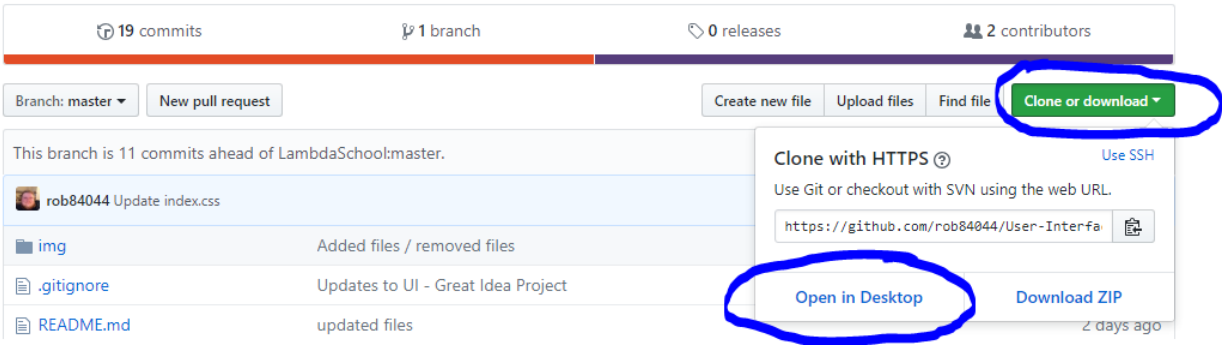
1. Fork the github repository



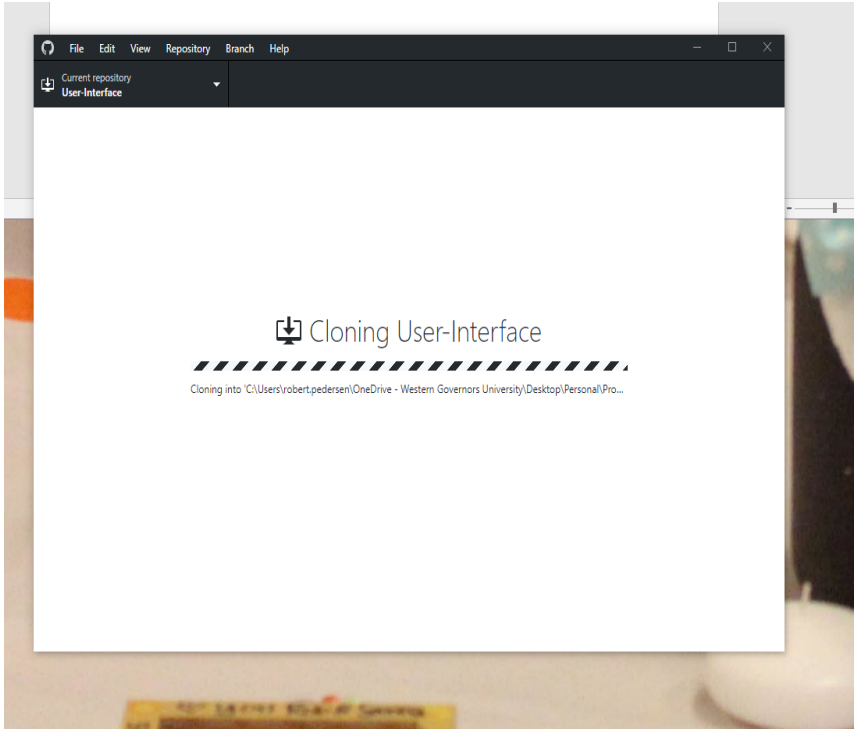
2. You can know it is your repository when you see your github name a slash and the name of the repository



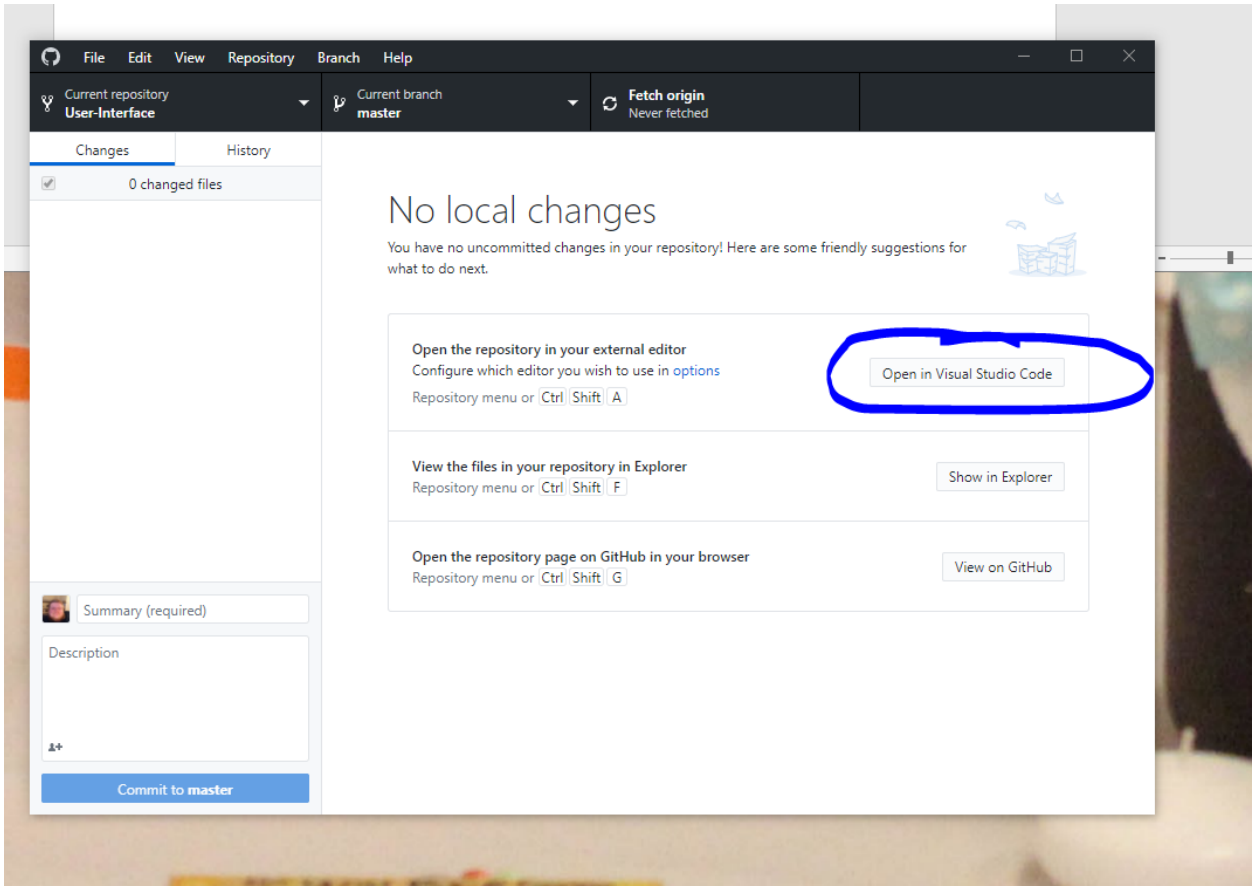
3. Click on clone or download green button on your newly forked repository.



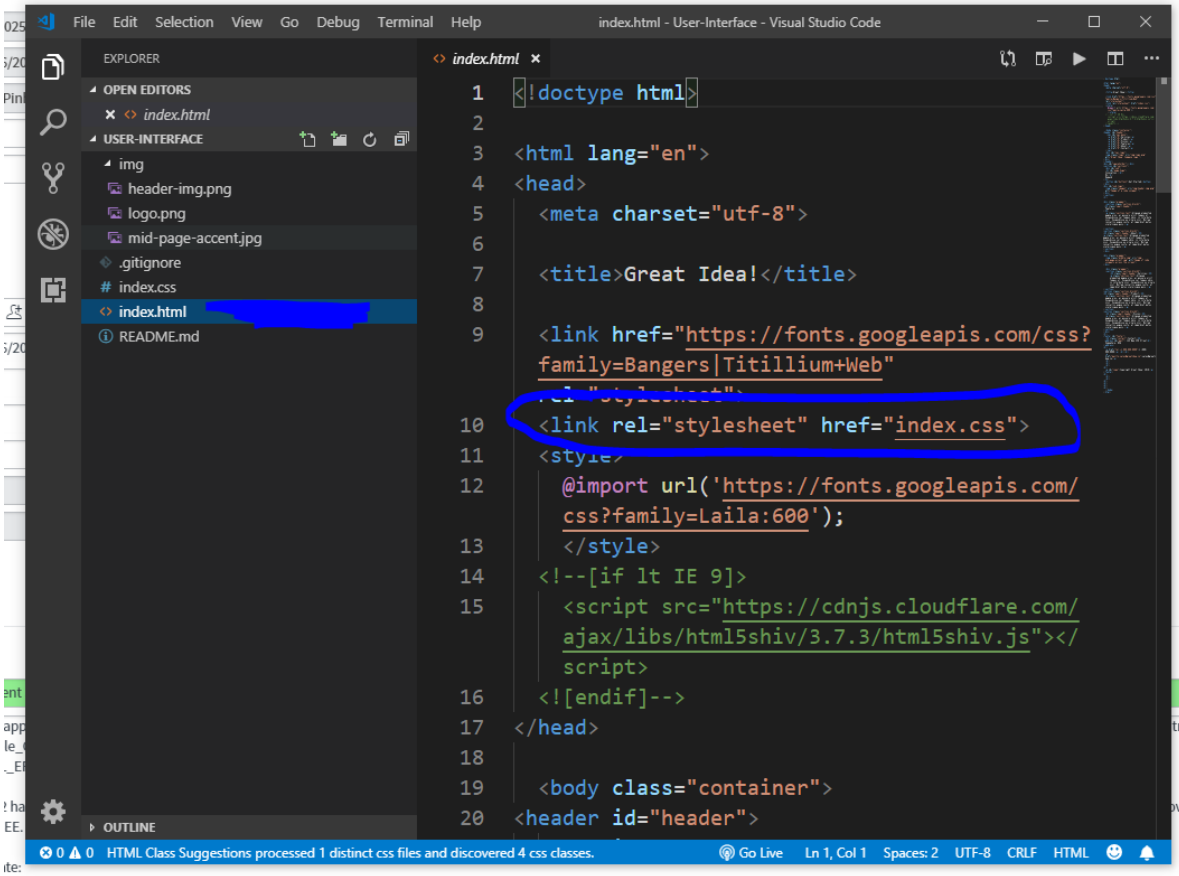
4. This will download the repository into the git hub desktop application



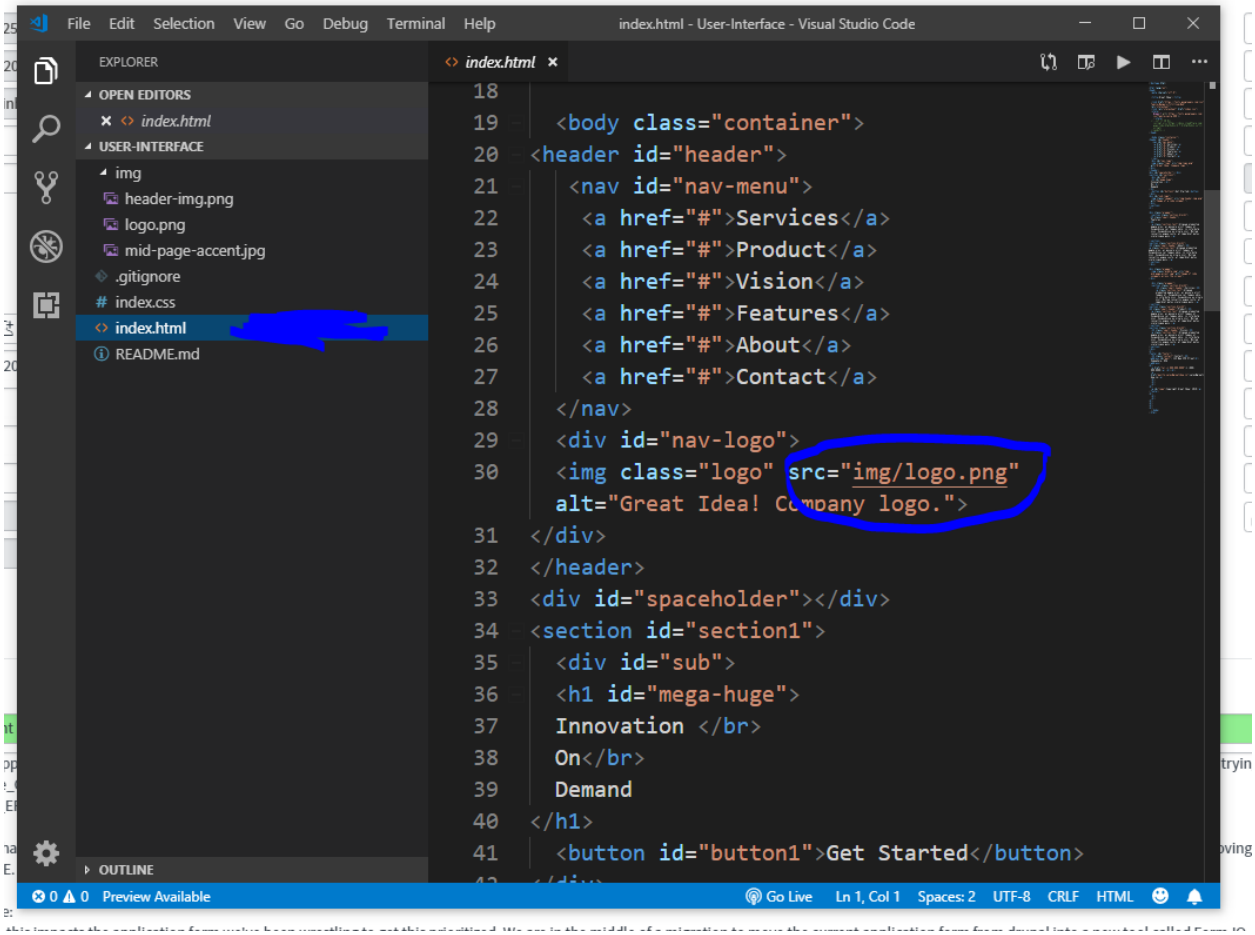
5. Once it is done being cloned, your git hub desktop app will look like this, select Open in Visual Studio Code



6. Once you do this, you will need to re-structure your file organization to your preferences. For myself, I like the html and CSS files to be in the root directory, and then I like to have image files 1 level deep (inside a folder that is in the root directory). However you do this, make sure you update your files and folders both in your file explorer and also adjust links in the actual files themselves

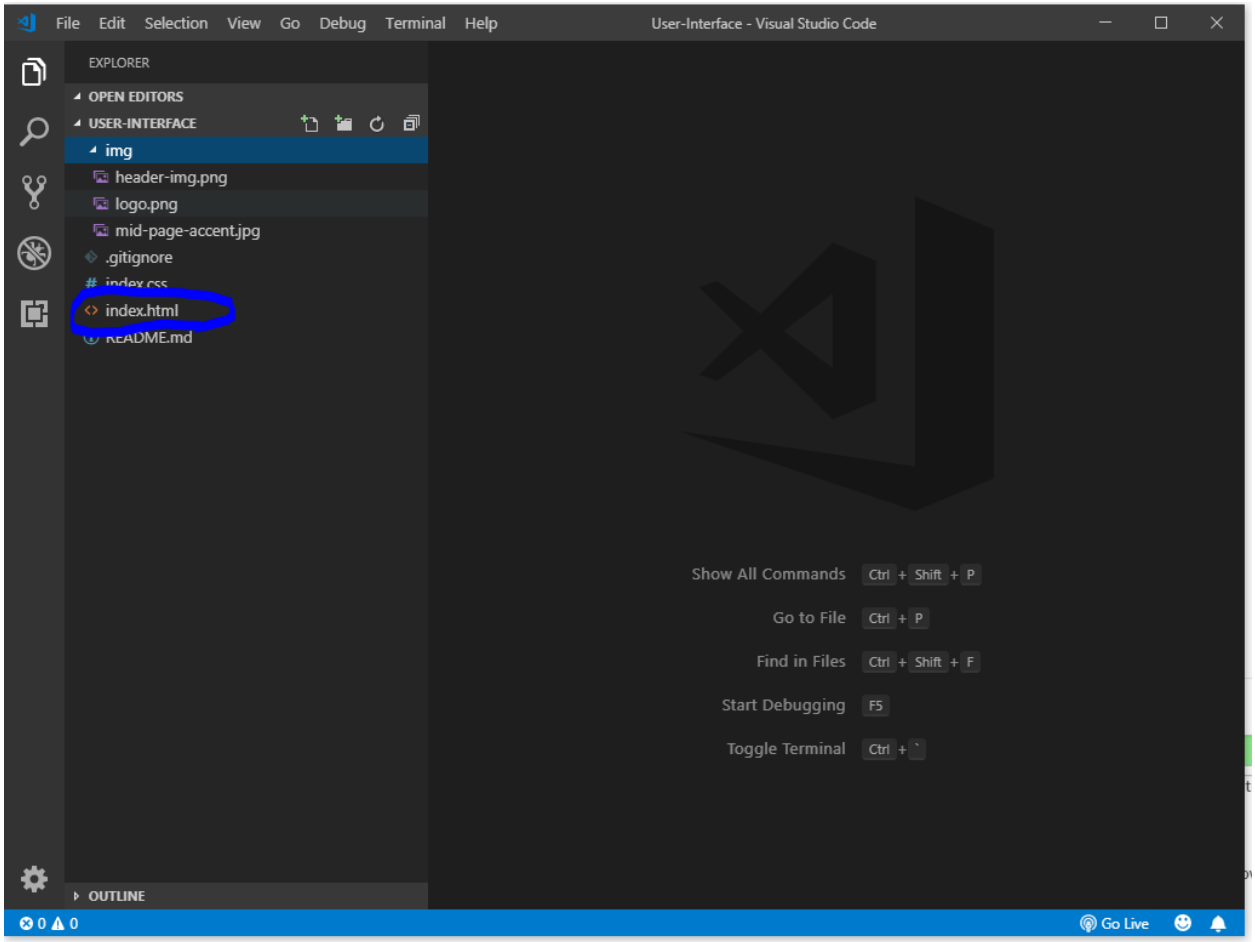


e this impacts the application form we've been wrestling to get this prioritized. We are in the middle of a migration to move the current application form from drupal into a new tool called Form.IO. We anti

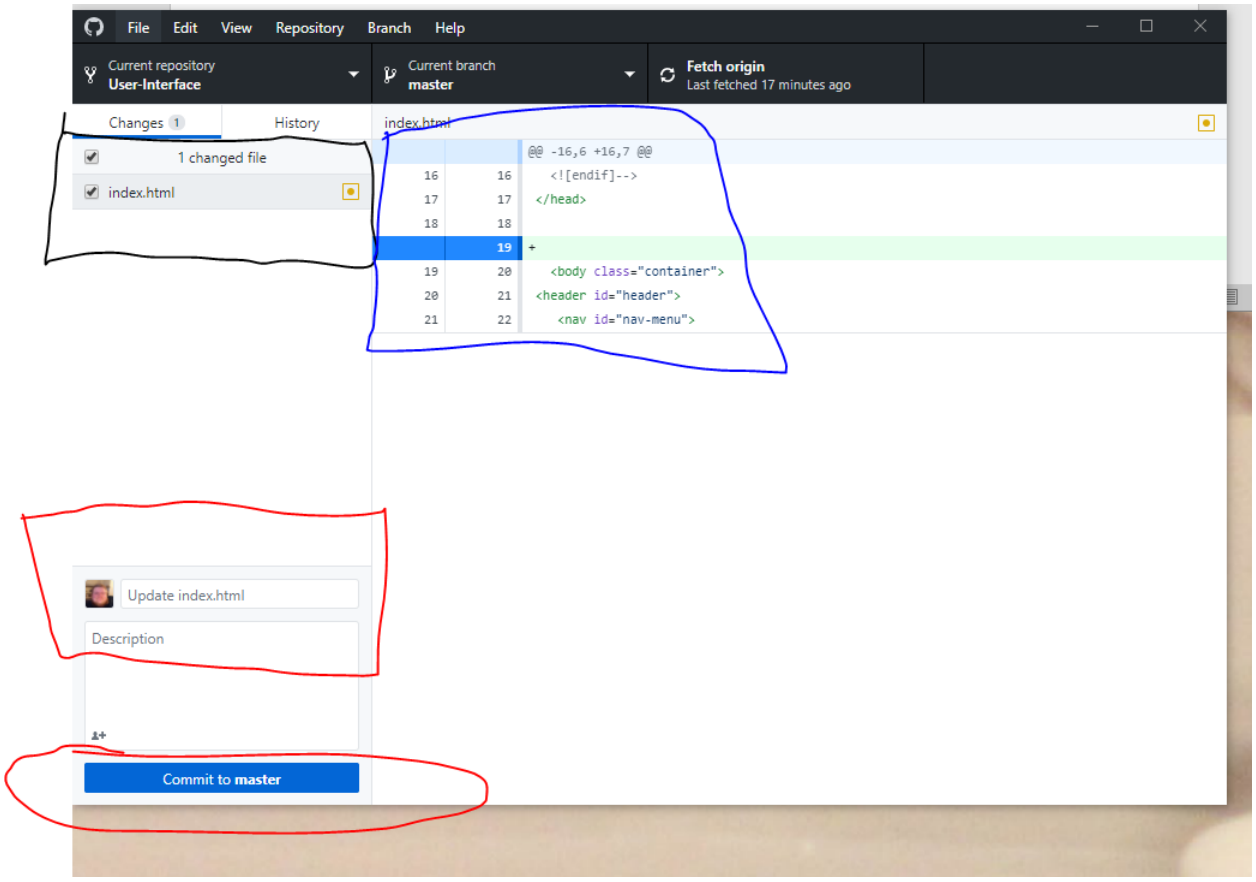


7. Once you do this, one of the extensions you should have installed already is called “Live server”. This allows you to see an automatically updated version of your site and updates each time you save something within Visual Studio code. To access this feature, right click the HTML file and select “Live

server".

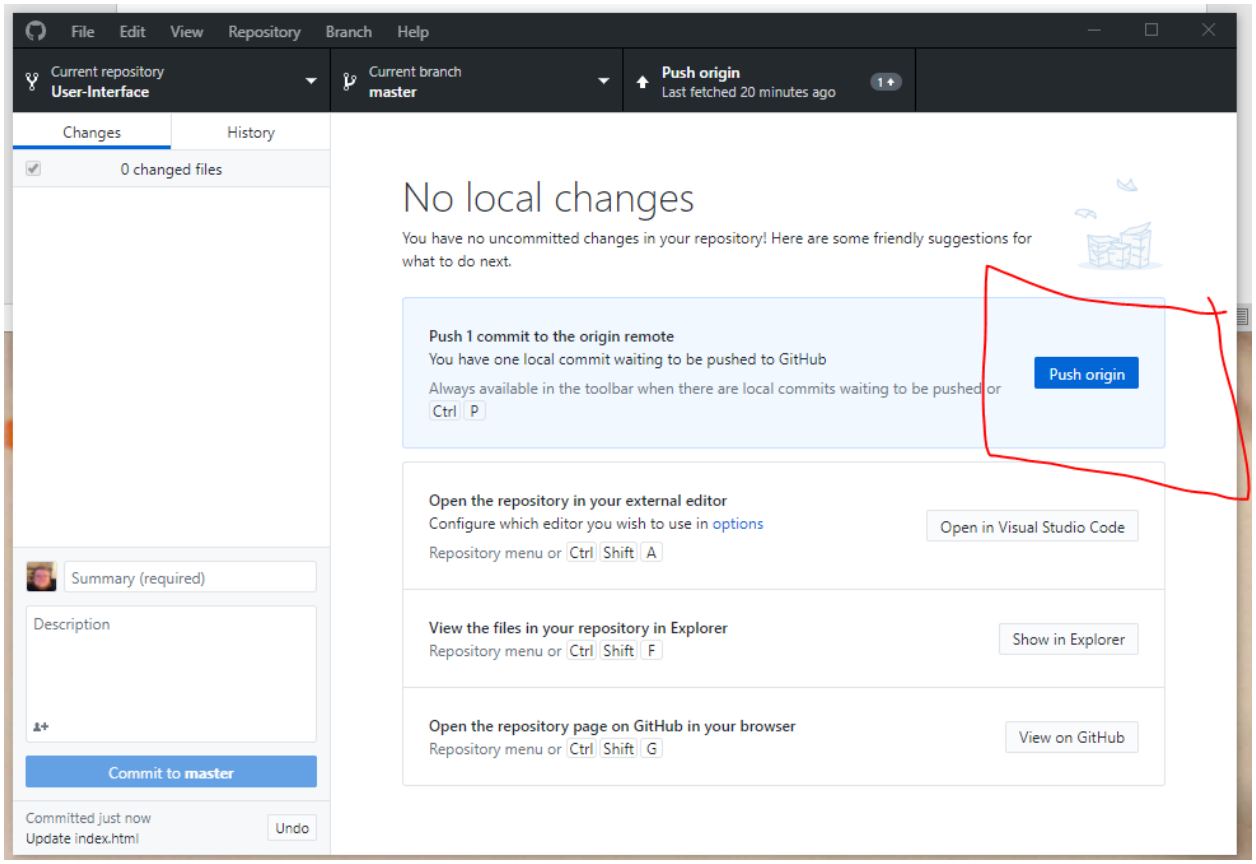


8. When you make a change to a file either in visual studio code or in file explorer, your git hub desktop app will auto recognize any changes. In the instance below it shows us what git hub desktop looks like when adding you add a new line to your file.
    a. Black box tells us what file(s) was changed
    b. Blue box tells us exactly what was changed or edited by using green for added things, and red for removed things
    c. The red boxes are important as they are what are describing the change that occurred that you are making as part of a permanent record for your repository. You must have something in the first small box, then press commit to master. This will prep the update to be pushed to the git hub website



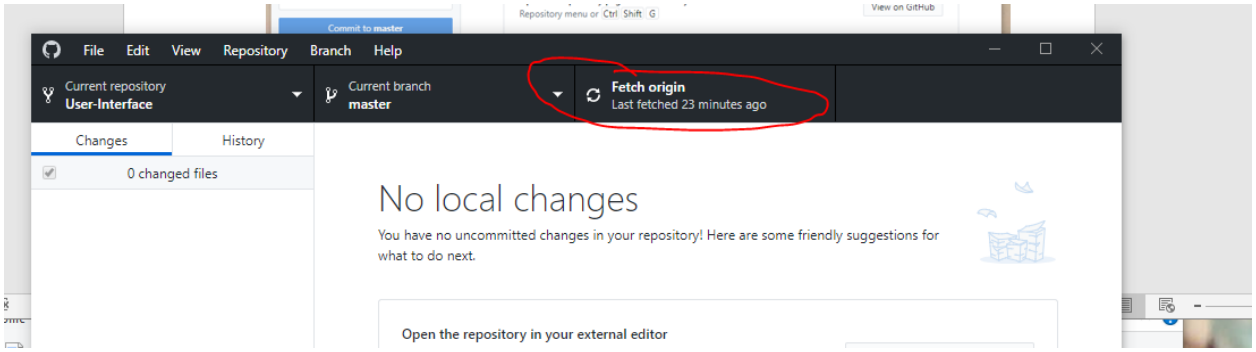
9. Once you hit commit, you are ready to push all of the information that was edited back to the git hub repository online. You will be taken to a page that looks like the below image and it will prompt you to hit “push origin” This will push your prepared commit to the online repository and changing it to match what your repository on your local computer looks like, essentially you are syncing the

online version, which is now outdated compared to your local version of the repository.

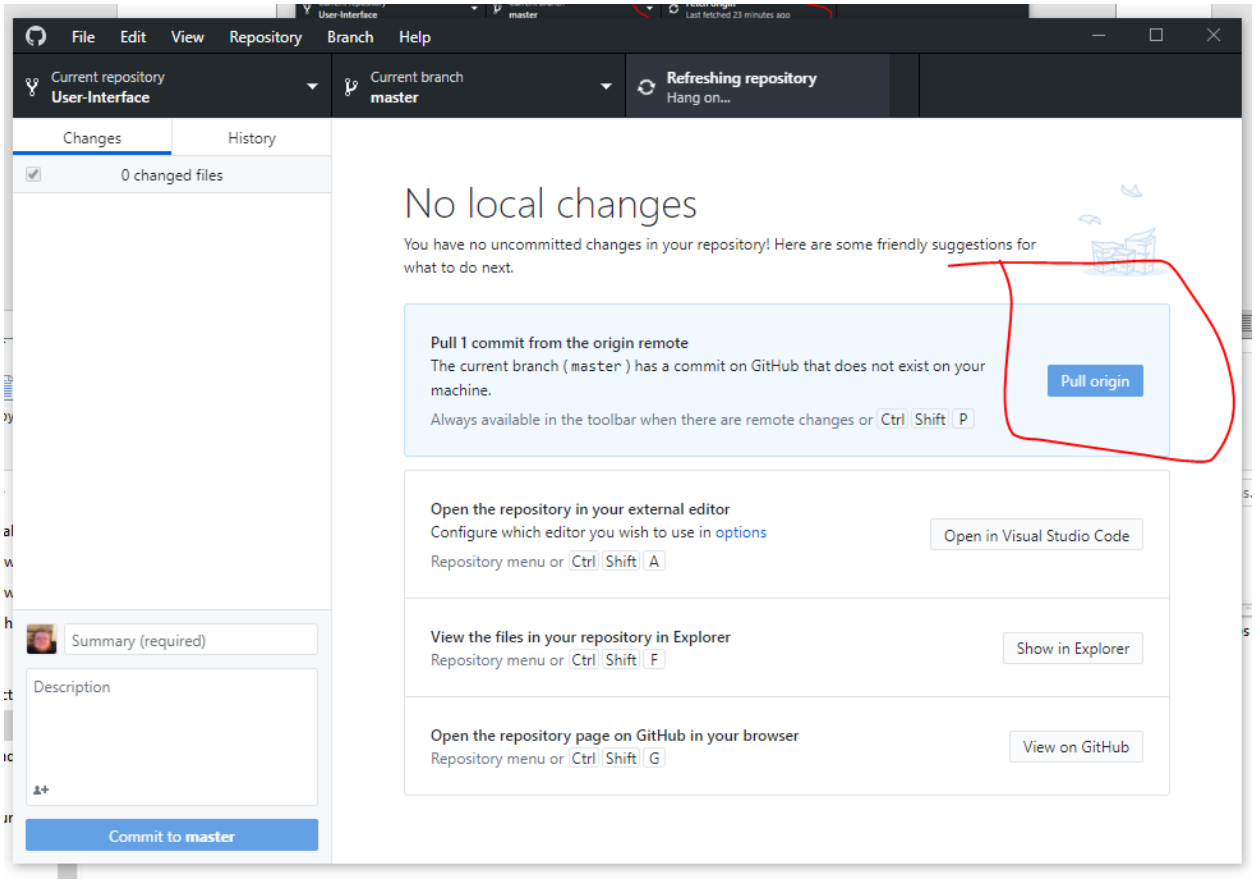


10. Congratulations! You have now completely synched your desktop repository copy with the online repository copy. Now lets say you have two computers. And the work above was done on computer 1 and not done on computer two. They both had the original copy of the repository downloaded to the computer, but only computer 1 made changes and synced with the git hub online repository. Well, then that means that computer 2 has an older version or outdated offline repository compared to the online repository. To make sure that computer 2, computer 1, and the online git hub website all have the same repository you will need to fetch the origin from the git hub website as shown below.

This step can also be done on any other computer that you want to work on your repository as well.



11. Once you click on fetch origin, if there are any updates on the git hub online repository you will see the image below. Make sure you select “pull origin” to update your offline repository.



12. Once you have an updated offline repository, you will see the below image. Now everything is all caught up!

You may share this as you see fit, please leave this piece of text here to give credit to the creator of me!

Created by: Robert Pedersen

Github handle: rob84044

Email robert@alambda.dev