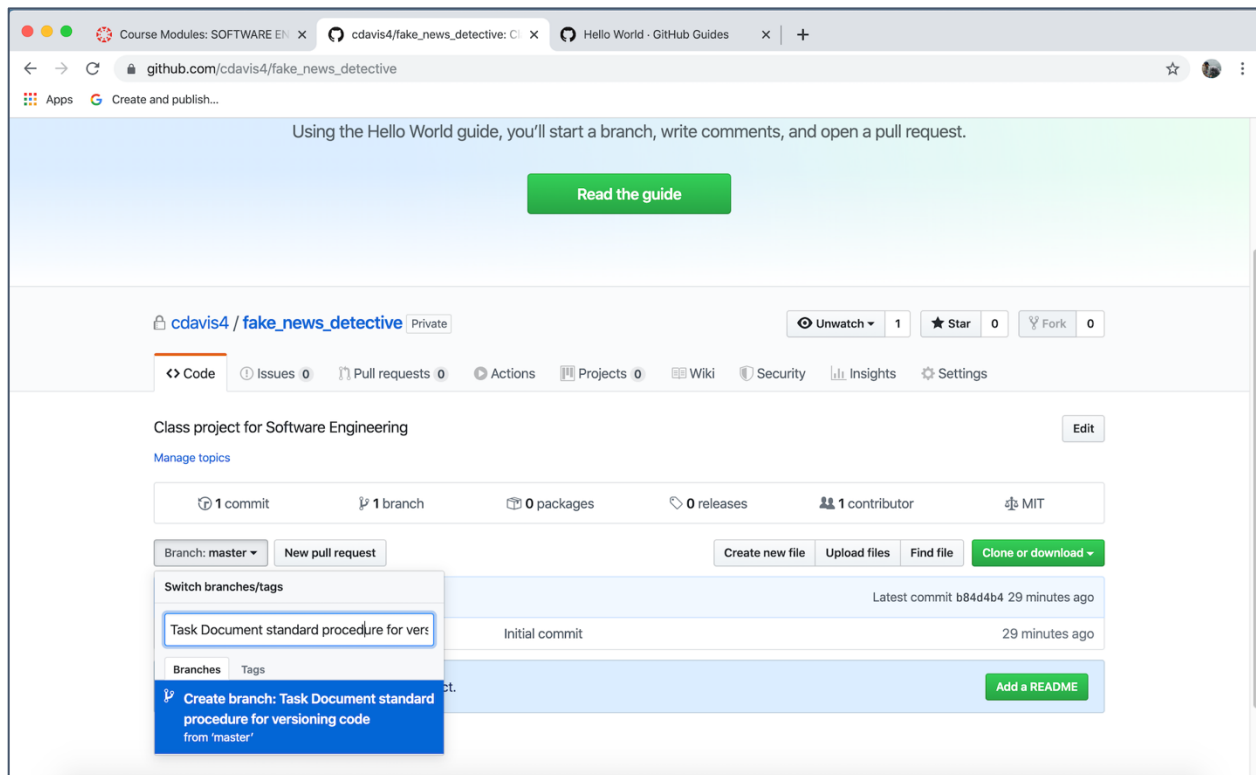


Group 30 Overview of Versioning Standards

repo https://github.com/cdavis4/fake_news_detective

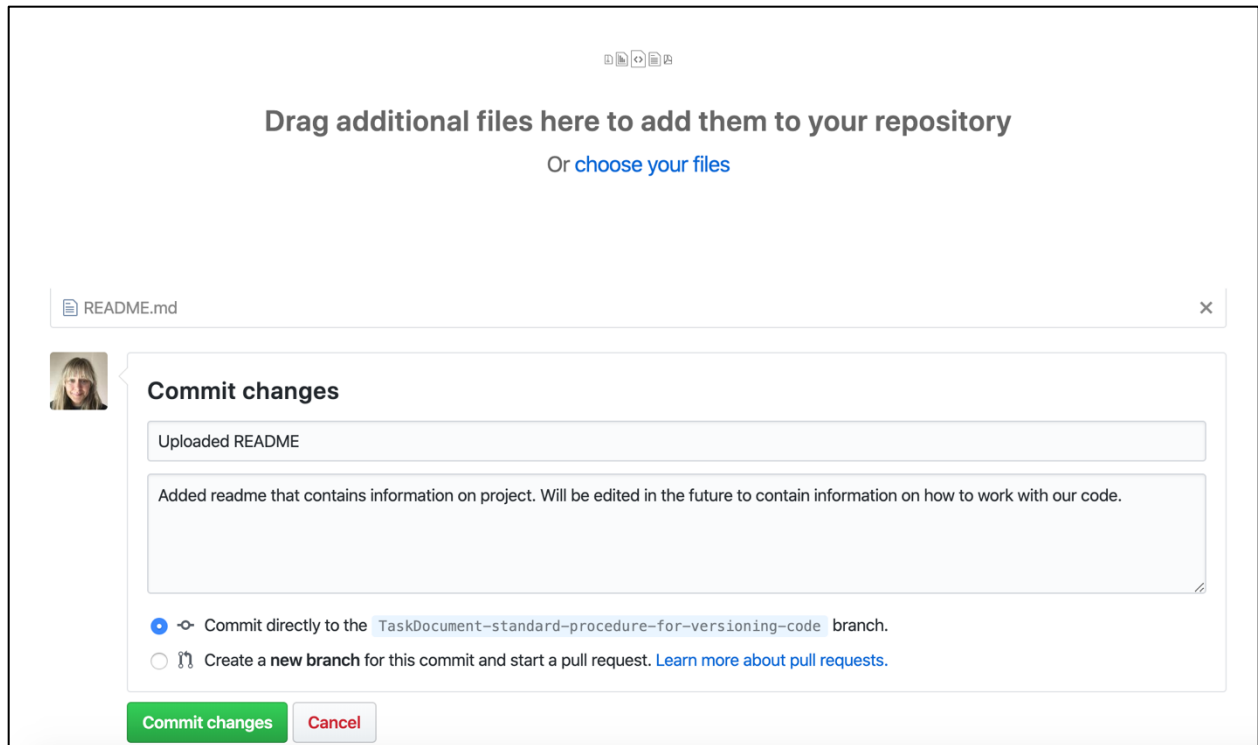
Create branch using the name of the task you are working on. Example: Document standard procedure for versioning code



Switch to your new branch this should happen automatically. This is the branch you will use to commit your edits to fulfill this specific task. **DO NOT COMMIT TO MASTER**

Commit Edits to this branch in any way you want. However, it's suggested to commit often as commits can be reverted if something terrible goes wrong. Below are some different methods to make these commits.

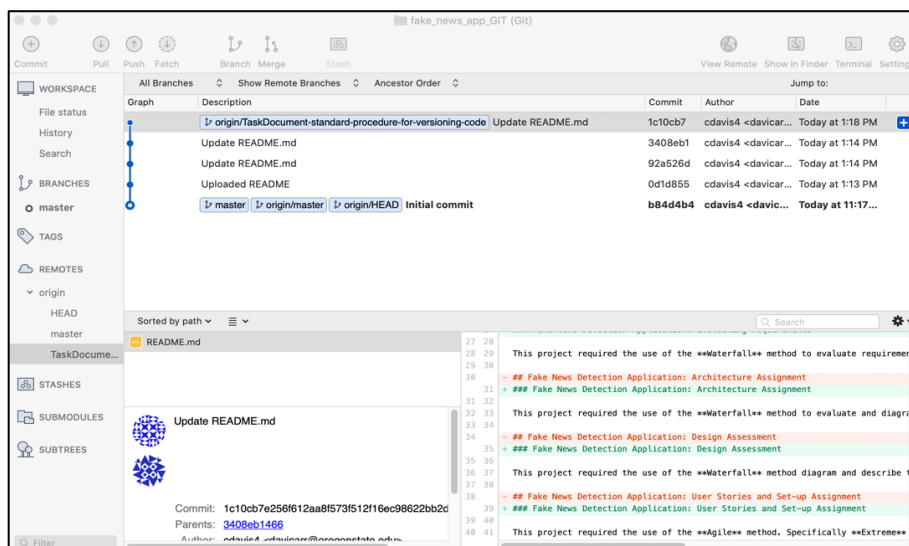
Upload Files in the web page of the repository when pointed to your branch. This will make a commit that overwrites existing files and/or adds new ones using a drag and drop approach. This is not recommended for complex edits but can be an easy way to add new files and data. Click Upload Files > drag and drop > Commit changes.



You may also install git and do this command line style.

<https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>

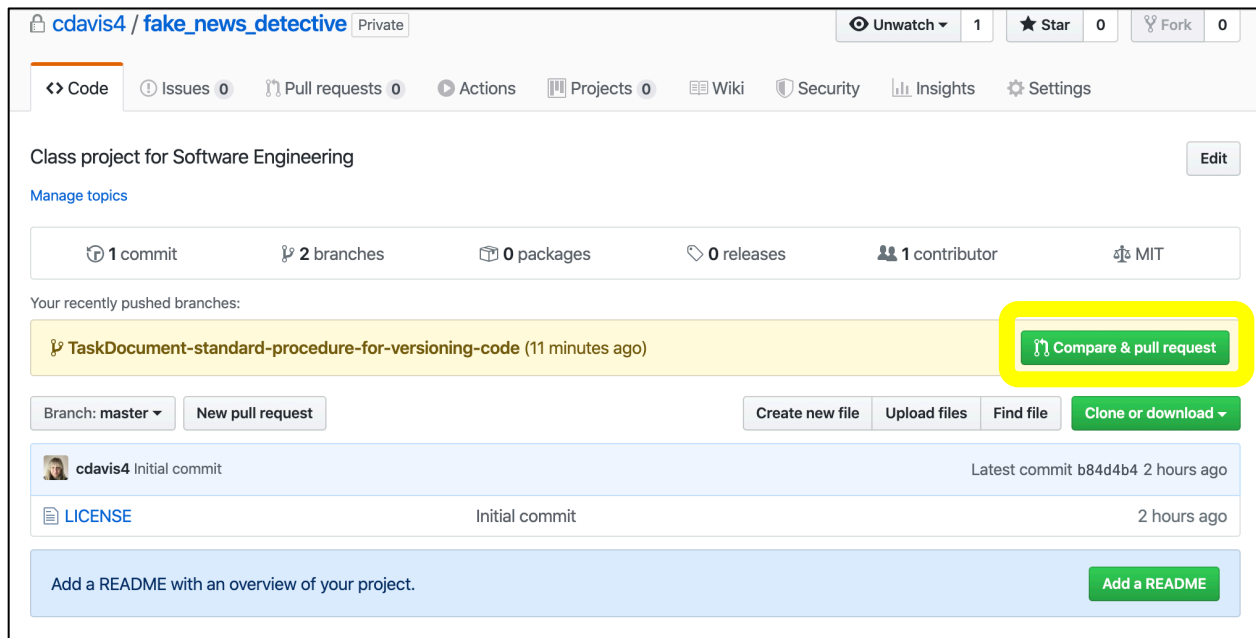
Use **SourceTree** it's more visual method, worth learning and also free. Download Free Version. <https://www.sourcetreeapp.com/>



Create pull request

If you are done with your task and feel good about your commit(s). You can now start the process of merging your code into the remote master branch (main working branch). Remember to fetch or pull updates from master before Compare & pull request to avoid conflicts with master branch. You need to keep your branch up to date with the other changes in master. Commits cannot be overwritten by pulls or fetches. However, you can overwrite or remove other teammate efforts by pushing or merging without thought into the master branch. Thank the GitHub God for revert.

Doc: <https://help.github.com/en/github/using-git/getting-changes-from-a-remote-repository>



Let your Pair Programmer review and Merge Pull Request

You can do this because you're an administrator, but this should be done by another teammate that can review your code. They can then approve changes and merge the pull request. You can make comments here too. If not ready to merge to let people know what you are trying to do. As a reviewer you can reject or comment on changes required before able to merge as well.

Delete your branch

Up to you but you prob. don't need it anymore. However, it might be good to keep around for prosperity.