

## Install Git

<https://git-scm.com/downloads>



## Install GitHub Desktop

<https://desktop.github.com/>

# Simple collaboration from your desktop

GitHub Desktop is a seamless way to contribute to projects on GitHub and GitHub Enterprise.

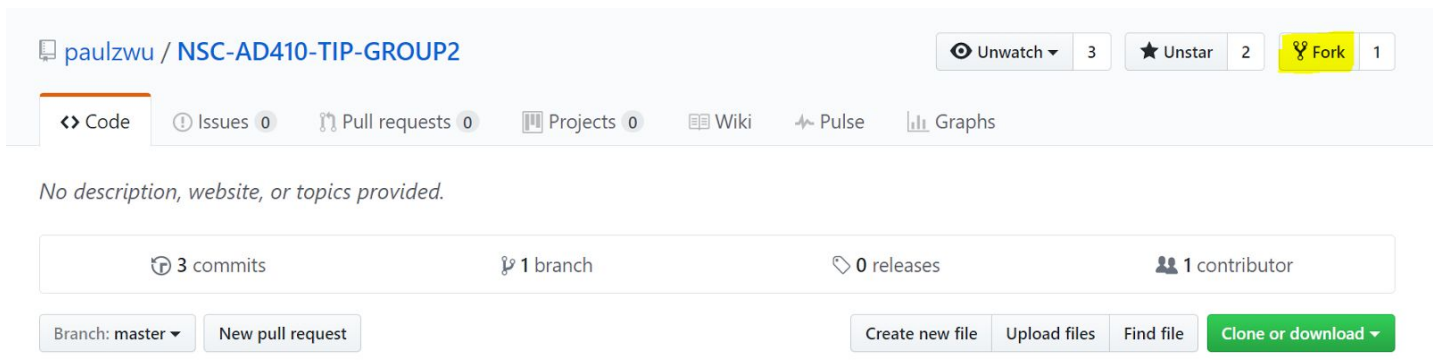
Available for Mac and [Windows](#)

Download GitHub Desktop  
Windows 7 or later

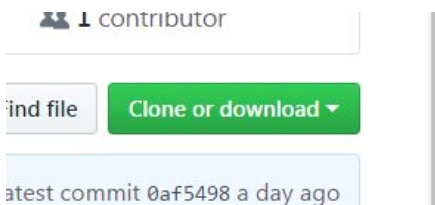
By clicking the Download button you agree to the  
End-User License Agreement

## Fork repository to your Github account

<https://github.com/paulzwu/NSC-AD410-TIP-GROUP2>



Clone your repository as zip file or copy link and clone from command line or GUI



Extract downloaded repository to GitHub folder on your system and import git project in Eclipse (repository needs to be initialized first...see command below)

## Commands

Clone your repo into GitHub folder (under Documents folder by default)

Git clone <https://github.com/yourGithubHandle/NSC-AD410-TIP-GROUP2.git>

```
C:\Users\just_\OneDrive\Documents\GitHub> git clone https://github.com/paulzwu/NSC-AD410-TIP-GROUP2.git
Cloning into 'NSC-AD410-TIP-GROUP2'...
remote: Counting objects: 12, done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 12 (delta 1), reused 8 (delta 0), pack-reused 0
Unpacking objects: 100% (12/12), done.
Checking connectivity... done.
```

Initialize repository (make sure you're in the project's directory)

Git init

```
posh-git ~ NSC-AD410-TIP-GROUP2-master [master]
C:\Users\just_\OneDrive\Documents\GitHub\NSC-AD410-TIP-GROUP2-master [master +5 ~0 -0 !]> git init
Reinitialized existing Git repository in C:/Users/just_/OneDrive/Documents/GitHub/NSC-AD410-TIP-GROUP2-master/.git/
```

Add remote repo to pull from master repo

Git remote add upstream <https://github.com/paulzwu/NSC-AD410-TIP-GROUP2.git>

```
C:\Users\just_\OneDrive\Documents\GitHub\NSC-AD410-TIP-GROUP2 [master ≡]> git remote -v
origin https://github.com/just255/NSC-AD410-TIP-GROUP2.git (fetch)
origin https://github.com/just255/NSC-AD410-TIP-GROUP2.git (push)
C:\Users\just_\OneDrive\Documents\GitHub\NSC-AD410-TIP-GROUP2 [master ≡]> git remote add upstream https://github.com/paulzwu/NSC-AD410-TIP-GROUP2.git
C:\Users\just_\OneDrive\Documents\GitHub\NSC-AD410-TIP-GROUP2 [master ≡]> git remote -v
origin https://github.com/just255/NSC-AD410-TIP-GROUP2.git (fetch)
origin https://github.com/just255/NSC-AD410-TIP-GROUP2.git (push)
upstream https://github.com/paulzwu/NSC-AD410-TIP-GROUP2.git (fetch)
upstream https://github.com/paulzwu/NSC-AD410-TIP-GROUP2.git (push)
C:\Users\just_\OneDrive\Documents\GitHub\NSC-AD410-TIP-GROUP2 [master ≡]>
```

Pull updated repository from project master

Git pull upstream master

```
C:\Users\just_\OneDrive\Documents\GitHub\NSC-AD410-TIP-GROUP2 [master ≡]> git pull upstream master
From https://github.com/paulzwu/NSC-AD410-TIP-GROUP2
 * branch      master      -> FETCH_HEAD
 * [new branch] master      -> upstream/master
Already up-to-date.
```

Create new branch from your repository

Git checkout -b NameOfBranch

```
C:\Users\just_\OneDrive\Documents\GitHub\NSC-AD410-TIP-GROUP2 [master ≡]> git checkout -b MyBranch
Switched to a new branch 'MyBranch'
C:\Users\just_\OneDrive\Documents\GitHub\NSC-AD410-TIP-GROUP2 [MyBranch]>
```

Switch branches

Git checkout NameOfBranch

```
C:\Users\just_\OneDrive\Documents\GitHub\NSC-AD410-TIP-GROUP2 [MyBranch]> git checkout master
Switched to branch 'master'
Your branch is up-to-date with 'origin/master'.
C:\Users\just_\OneDrive\Documents\GitHub\NSC-AD410-TIP-GROUP2 [master ≡]>
```

## Make your edits to local files

### Status of branch

*Git status (Files you've edited are shown to not be staged yet-see below)*

Files in red have not been staged  
←

```
C:\Users\just_\OneDrive\Documents\GitHub\NSC-AD410-TIP-GROUP2 [master ==]> git status
On branch master
Your branch is up-to-date with 'origin/master'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)

    .metadata/
    .project
    status_report/Justin Simmons/

nothing added to commit but untracked files present (use "git add" to track)
```

### Stage files

*Git add \* or git add filename (hint: Press tab after typing 'git add' to cycle files)*

```
C:\Users\just_\OneDrive\Documents\GitHub\NSC-AD410-TIP-GROUP2 [master == +3 ~0 -0 !]> git add *
warning: LF will be replaced by CRLF in .metadata/.log.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in .metadata/.plugins/org.eclipse.epp.logging.aeri.ide/org.e
.ide.server/server-config.json.
The file will have its original line endings in your working directory.
warning: LF will be replaced by CRLF in .metadata/.plugins/org.eclipse.m2e.logback.configuration/
1933.xml.
The file will have its original line endings in your working directory.
C:\Users\just_\OneDrive\Documents\GitHub\NSC-AD410-TIP-GROUP2 [master == +63 ~0 -0 ~]>
```

### Commit staged files to your local machine

*Git commit -m "Type a brief description"*

```
C:\Users\just_\OneDrive\Documents\GitHub\NSC-AD410-TIP-GROUP2 [master == +63 ~0 -0 ~]> git commit -m "Initial commit"
[master ae94dcd] Initial commit
```

### Switch to master branch

*git checkout master*

### Merge branch into master

*Git merge nameOfBranch (make sure you're in master branch)*

### Push changes to your github repository

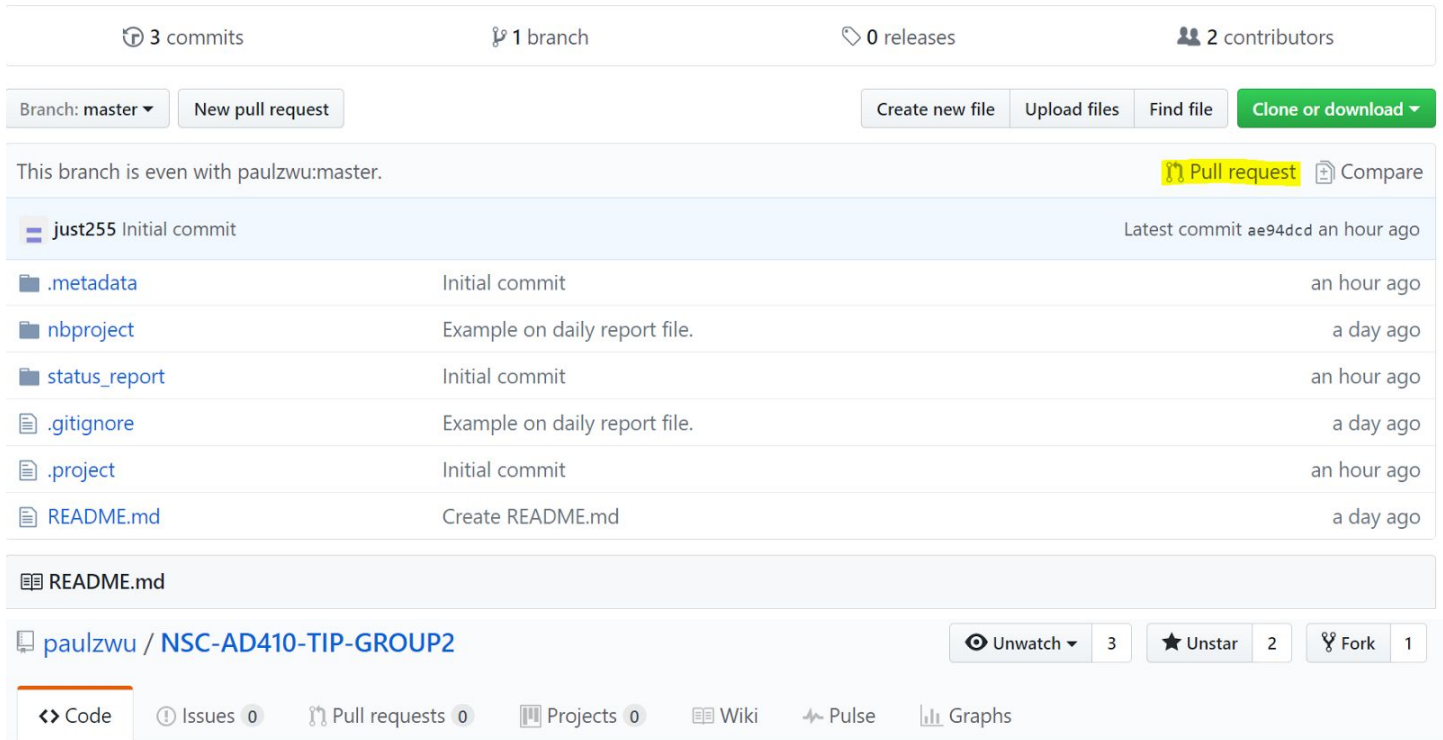
*Git push origin master*

```
C:\Users\just_\OneDrive\Documents\GitHub\NSC-AD410-TIP-GROUP2 [master !]> git push origin master
Counting objects: 75, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (58/58), done.
Writing objects: 100% (75/75), 51.93 KiB | 0 bytes/s, done.
Total 75 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/paulzwu/NSC-AD410-TIP-GROUP2.git
  0af5498..ae94dcd master -> master
```

## GitHub

Send your code up for review (pull request)

Go to your Github account and find the project's repo and click pull request



## Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).



Only one pull request is needed until your code is committed to the master. For example, if your code is reviewed and needs changes, you only need to commit and push your changes to your github. The changes will show in the pull request. Once your pull request is committed to the master, you will need to create a new one to send more code for review.

## Syncing a fork

from: <https://help.github.com/articles/syncing-a-fork/>

Sync a fork of a repository to keep it up-to-date with the upstream repository.

Before you can sync your fork with an upstream repository, you must [configure a remote that points to the upstream repository](#) in Git.

1. Open Terminal or Git Bash
2. Change the current working directory to your local project.
3. Fetch the branches and their respective commits from the upstream repository. Commits to `master` will be stored in a local branch, `upstream/master`.
4. `git fetch upstream`



```

remote: Counting objects: 75, done.
remote: Compressing objects: 100% (53/53), done.
remote: Total 62 (delta 27), reused 44 (delta 9)
Unpacking objects: 100% (62/62), done.
From https://github.com/ORIGINAL_OWNER/ORIGINAL_REPOSITORY
* [new branch]      master    -> upstream/master

```

5. Check out your fork's local `master` branch.
6. `git checkout master`  
Switched to branch 'master'
7. Merge the changes from `upstream/master` into your local `master` branch. This brings your fork's `master` branch into sync with the upstream repository, without losing your local changes.
8. `git merge upstream/master`  
Updating a422352..5fdff0f  
Fast-forward  

```

README          |    9 -----
README.md       |    7 ++++++
2 files changed, 7 insertions(+), 9 deletions(-)
delete mode 100644 README
create mode 100644 README.md

```
9. If your local branch didn't have any unique commits, Git will instead perform a "fast-forward":
10. `git merge upstream/master`  
Updating 34e91da..16c56ad  
Fast-forward  

```

README.md       |    5 +++--
1 file changed, 3 insertions(+), 2 deletions(-)

```

Tip: Syncing your fork only updates your local copy of the repository. To update your fork on GitHub, you must [push your changes](#).

