Ok so all of this is based upon a bash shell. If you are running Windows and don't have the bash shell I suggest either (A) get it, (B) start running a linux VM, (C) get the github desktop client (I can't help much with how to use that), or (D) google the equivalent commands.

Step 1: Make sure you have a github account. Go to www.github.com and login. Then send me (Charles) your email or username on discord

Step 2: Decide where you want your project to live on your system. In your terminal go to that location and enter the command:

```
git clone -b develop https://github.com/zaharacw/CSCD429_Project.git
```

You now have the repository on your system.

Step 3: Navigate to your home directory (cd ~). Check and see if you have a .gitconfig file (ls -al). If you do then check to see if the user section is configured with your name and email (cat .gitconfig). If it is then great! If it is not, or if you don't have a .gitconfig at all then follow the next steps:

```
cd path/to/where/you/cloned/CSCD429_Project/
cp gitconfig_example.txt ~/
```

Open the gitconfig_example.txt and edit the user name and email information to your own info. The rest of the stuff is just some git configurations that I have, if you want them keep them, if not then delete the rest of the line

```
mv gitconfig_example.txt ~/.gitconfig
```

Step 4: Test that it is all working

```
touch testFile.yourname.txt
git add testFile.yourname.txt
git commit -m "Testing commit of (Your name)"
git push
```

Go to github and make sure it worked!

Additional stuff

SSH Key

Do you think entering passwords is for chumps? Then SSH keys are for you!

Step 1: First check if you already have a public key

```
cd ~/.ssh
ls -al
```

If there is an id_rsa.pub then skip ahead to Step 3

Step 2: Create your public key, the "identifier" is how you want this public key to be identified. Typically use an identifying username, or your system. I've got a bunch on different linux boxes so I went with zaharacw-cscd429 (note I have things like zaharacw-mbp2017-personal as other keys to identify user-computer-profile)

```
ssh-keygen -t rsa -N "" -C "identifier"
```

Step 3: Copy your public key into github

```
cd ~/.ssh
cat id_rsa.pub
```

Copy the terminal contents. Go to github.com, upper right hand corner is your avatar, click that and select “settings”. Then on the left hand navigation select “SSH and GPG keys” then “new SSH key”. Paste the key you copied from the “cat” command into the key section on this page. You should not need to enter a title, it will parse it from your key because you put the identifier.

Next time you push to github you won’t need to enter your username or password!!

Basic Git Flow

A quick suggestion on Git flows. I had you clone “develop” so that we can keep “master” clean in case of severe conflicts or git no-nos. There are 2 main ways you can choose to work in git. Method 1, make all your changes on “develop,” commit to and push straight to “develop.” This will work, but the likelihood that you end up with conflicts increases dramatically. So here is what I would recommend.

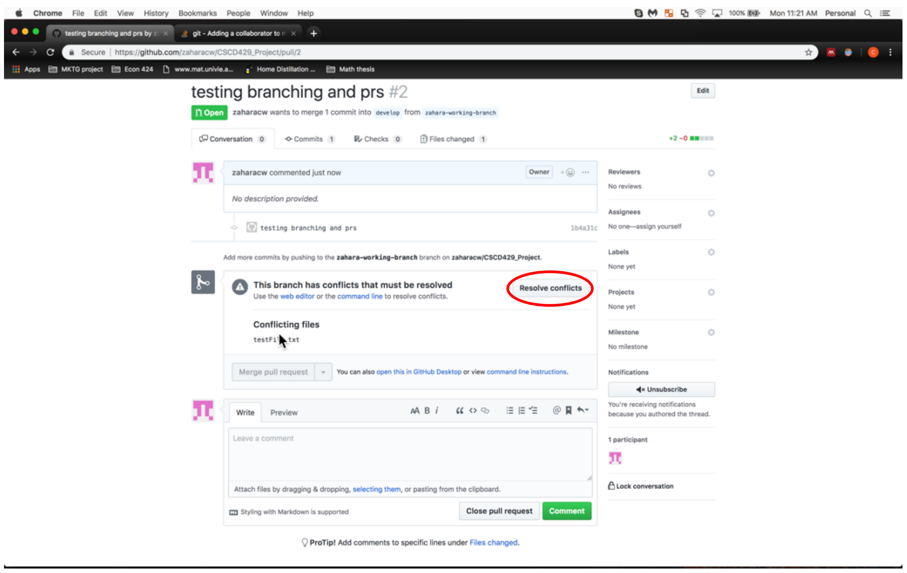
Step 1: create a personal branch off develop

```
git checkout -b your-branch-name develop
```

Step 2: Now do all your work on that branch as normal. Git add, commit, push.

Step 3: When you are ready to have other people see/work with your code then go to github.com and create a pull request to develop. Every Sunday night I will make a pull request from develop to master so that we have our master branch up to date.

Step 4: If you have conflicts in your PR from your personal branch to develop then resolve either in github or locally. Git hub method is to simply select “resolve conflicts.” This will bring up an online editor and you must resolve the issues there. (Be careful)


testing branching and prs #2

Open zaharacw wants to merge 1 commit into `develop` from `zahara-working-branch`

Conversation 0 | Commits 1 | Checks 0 | Files changed 1 | +2 −0

zaharacw commented just now — Owner

No description provided.

`testing branching and prs` 1b4a31c

Add more commits by pushing to the **zahara-working-branch** branch on **zaharacw/CSCD429_Project**.

This branch has conflicts that must be resolved
Use the web editor or the command line to resolve conflicts.

Resolve conflicts

Conflicting files
`testFi txt`

Merge pull request — You can also open this in GitHub Desktop or view command line instructions.

Write | Preview

Leave a comment

Attach files by dragging & dropping, selecting them, or pasting from the clipboard.

Styling with Markdown is supported — Close pull request — Comment

ProTip! Add comments to specific lines under Files changed.

Reviewers: No reviews
Assignees: No one—assign yourself
Labels: None yet
Projects: None yet
Milestone: No milestone
Notifications: Unsubscribe — You're receiving notifications because you authored the thread.
1 participant
Lock conversation

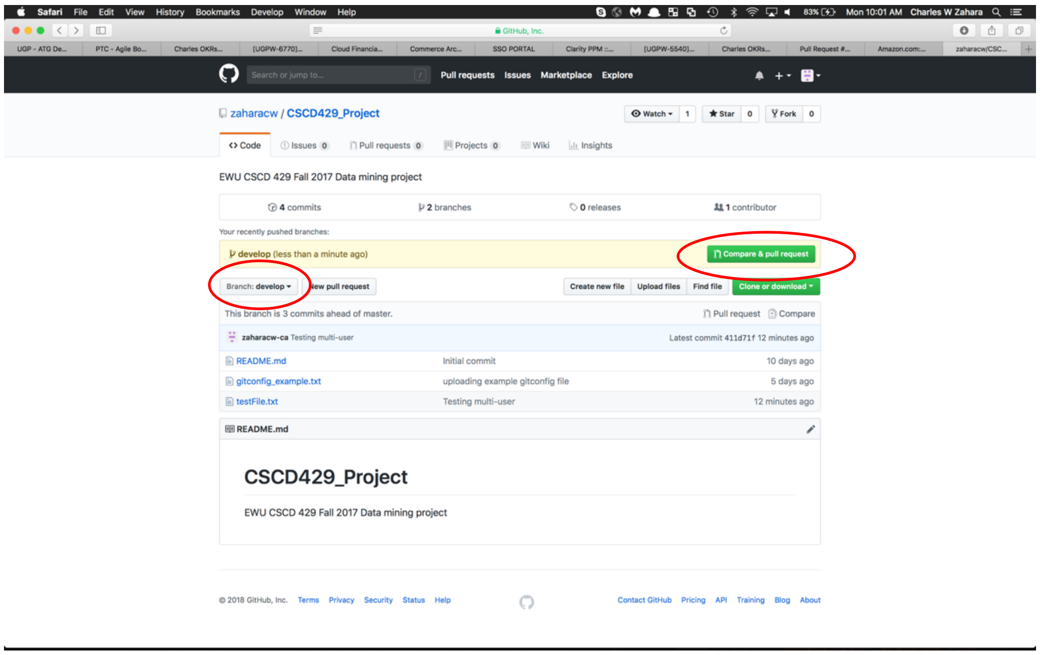
Command line method is:

```
git checkout develop
git pull
git checkout your-branch-name
git merge develop
```

You will have conflicts. Look at the code and resolve the conflicts. I usually use “git mergetool” which is a UI merging assistant, but it takes some practice to get used to. (Ask for help if you need help). Make sure all your files are added back to the commit (check using git status). Once all files are added complete your merge using git commit (I think? Check the instructions from git status). Then git push. Now your PR won’t be in conflict.

Pull Requests

Pull requests are essentially a monitored merge. To create a pull request got to github.com and to our project repository. Then select the branch you want to merge FROM (probably your local branch). You should see a yellow bar that has a “compare and pull request” on the right hand side. Select “compare and pull request”.


zaharacw / CSCD429_Project

EWU CSCD 429 Fall 2017 Data mining project

4 commits · 2 branches · 0 releases · 1 contributor

Your recently pushed branches:

develop (less than a minute ago) — Compare & pull request

Branch: develop · New pull request · Create new file · Upload files · Find file · Clone or download

This branch is 3 commits ahead of master. · Pull request · Compare

zaharacw-ca Testing multi-user · Latest commit 411d71f 12 minutes ago

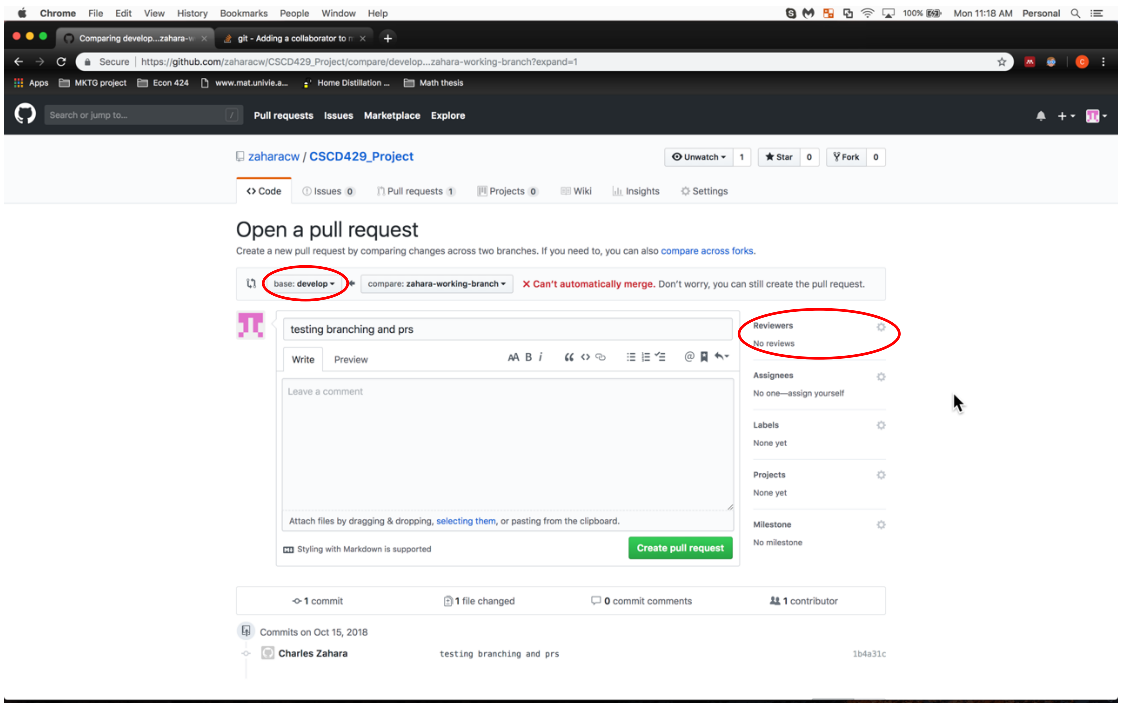
README.md	Initial commit	10 days ago
gitconfig_example.txt	uploading example gitconfig file	5 days ago
testFile.txt	Testing multi-user	12 minutes ago

README.md

CSCD429_Project

EWU CSCD 429 Fall 2017 Data mining project

Now make sure you have the correct branch selected to merge TO (probably develop). On the right hand side you will see “Reviewers” select that and add everyone. This way everyone can see what you changed and when you changed it.


zaharacw / CSCD429_Project

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also compare across forks.

base: develop · compare: zahara-working-branch · ✕ Can't automatically merge. Don't worry, you can still create the pull request.

testing branching and prs

Leave a comment

Attach files by dragging & dropping, selecting them, or pasting from the clipboard.

Styling with Markdown is supported · Create pull request

Reviewers: No reviews

Assignees: No one—assign yourself

Labels: None yet

Projects: None yet

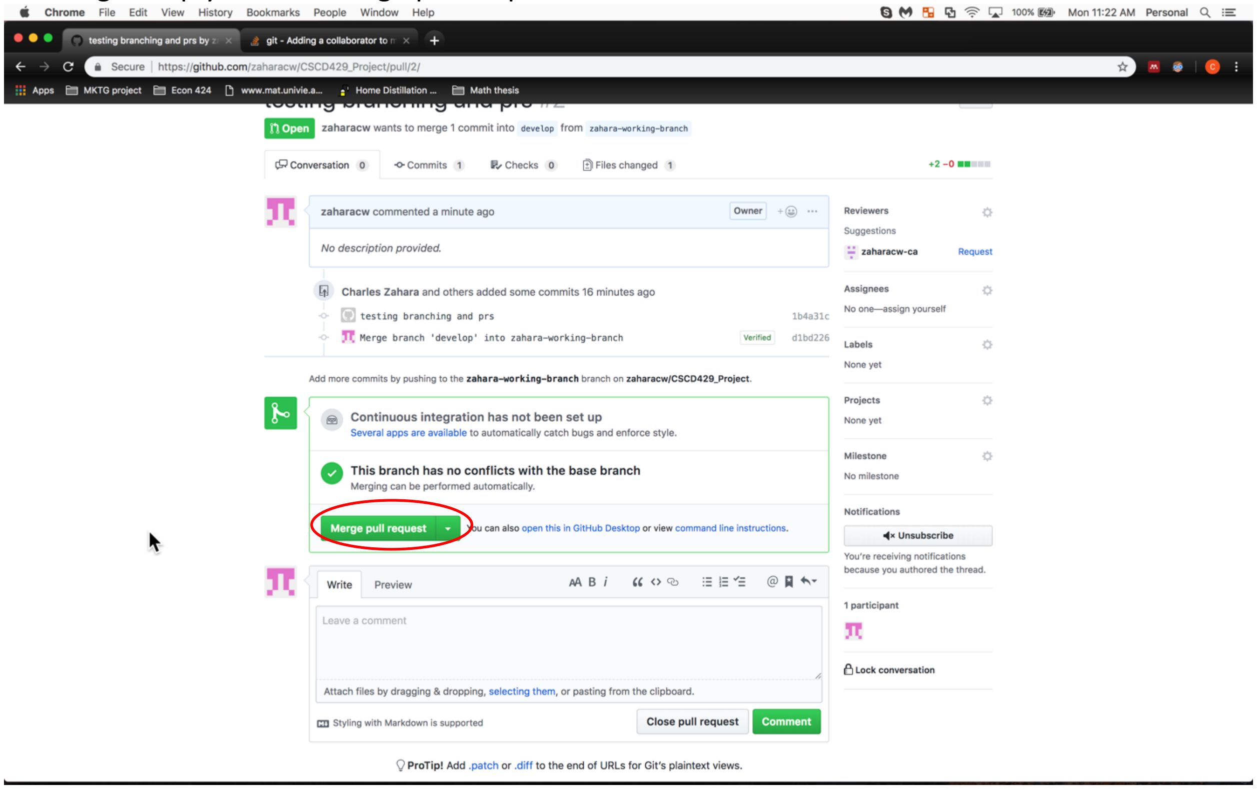
Milestone: No milestone

1 commit · 1 file changed · 0 commit comments · 1 contributor

Commits on Oct 15, 2018

Charles Zahara · testing branching and prs · 1b4a31c

Select “create pull request.” At this point it is up to everyone to decide what we want to do. We can require/request that people actually review code before we merge it in, OR we can just let anyone merge their own stuff. Currently it is set up so that anyone can merge their own stuff. To merge simply select “Merge pull request”



https://github.com/zaharacw/CSCD429_Project/pull/2/

Open zaharacw wants to merge 1 commit into develop from zahara-working-branch

zaharacw commented a minute ago

No description provided.

Charles Zahara and others added some commits 16 minutes ago

testing branching and prs 1b4a31c

Merge branch 'develop' into zahara-working-branch Verified d1bd226

Add more commits by pushing to the **zahara-working-branch** branch on **zaharacw/CSCD429_Project**.

Continuous integration has not been set up
Several apps are available to automatically catch bugs and enforce style.

This branch has no conflicts with the base branch
Merging can be performed automatically.

You can also open this in GitHub Desktop or view command line instructions.

Attach files by dragging & dropping, selecting them, or pasting from the clipboard.

Styling with Markdown is supported

Reviewers: Suggestions zaharacw-ca Request
Assignees: No one—assign yourself
Labels: None yet
Projects: None yet
Milestone: No milestone
Notifications: You're receiving notifications because you authored the thread.
1 participant

ProTip! Add .patch or .diff to the end of URLs for Git's plaintext views.