

Git Forking Workflow Lab

Pre-requisites

1. Sign up for [GitHub](https://github.com/) [https://github.com/].
2. [Install](https://linode.com/docs/development/version-control/how-to-install-git-on-linux-mac-and-windows/) [https://linode.com/docs/development/version-control/how-to-install-git-on-linux-mac-and-windows/] `git` locally in your development virtual machine.
3. Setup SSH key for GitHub.
 - a. [Generate SSH key to use with git.](https://help.github.com/articles/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent/)
[https://help.github.com/articles/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent/]
 - b. [Add SSH key to your github account.](https://help.github.com/articles/adding-a-new-ssh-key-to-your-github-account/)
[https://help.github.com/articles/adding-a-new-ssh-key-to-your-github-account/]
4. Run thru initial setup for git.
 - a. [Setting your username in Git](https://help.github.com/articles/setting-your-username-in-git/#platform-linux) [https://help.github.com/articles/setting-your-username-in-git/#platform-linux]
 - b. [Setting your commit email address in Git](https://help.github.com/articles/setting-your-commit-email-address-in-git/#platform-linux)
[https://help.github.com/articles/setting-your-commit-email-address-in-git/#platform-linux]



Reading

1. [Forking Workflow](https://www.atlassian.com/git/tutorials/comparing-workflows/forking-workflow) [https://www.atlassian.com/git/tutorials/comparing-workflows/forking-workflow].

2. [Collaborative development models](https://help.github.com/articles/about-collaborative-development-models/) [https://help.github.com/articles/about-collaborative-development-models/]
3. [Pull Requests](https://help.github.com/articles/about-pull-requests/) [https://help.github.com/articles/about-pull-requests/]

Setup GitHub Repositories for Collaboration



1. Choose one person in your team to be the “scrum master”.

2.  **Scrum Master should now create a repository in their GitHub called 'csye6225-git-forking-workflow'.** 

Create a new repository


A repository contains all the files for your project, including the revision history.


Owner **Repository name**

 **tejasparikh** / **csye6225-git-forking-workflow** 




Great repository names are short and memorable. Need inspiration? How about **vigilant-rotary-phone**.

Description (optional)



☐  **Public**
Anyone can see this repository. You choose who can commit.

☒  **Private**
You choose who can see and commit to this repository.

☐ **Initialize this repository with a README** **Don't add anything to repo**
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

 **Add .gitignore:** **None**  **Add a license:** **None** 

Create repository

3.  **Import code from the starter application at** 

<https://github.com/tejasparikh/csye6225-git-forking-workflow>

[\[https://github.com/tejasparikh/csye6225-git-forking-workflow\]](https://github.com/tejasparikh/csye6225-git-forking-workflow)

a.

tejasparikh / csye6225-git-forking-workflow Private

Watch 0 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH git@github.com:tejasparikh/csye6225-git-forking-workflow.git

We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# csye6225-git-forking-workflow" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin git@github.com:tejasparikh/csye6225-git-forking-workflow.git
git push -u origin master
```

...or push an existing repository from the command line

```
git remote add origin git@github.com:tejasparikh/csye6225-git-forking-workflow.git
git push -u origin master
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Import code

b.

Import your project to GitHub

Import all the files, including the revision history, from another version control system.

Your old repository's clone URL

https://github.com/tejasparikh/csye6225-git-forking-workflow.git

Learn more about the types of [supported VCS](#).

Your existing repository

tejasparikh/

Change repository

Cancel



Begin Import

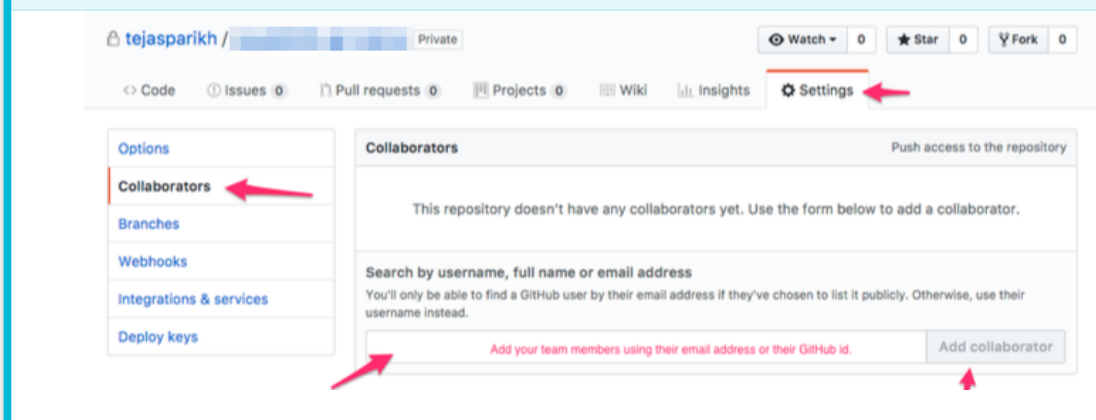
c.



Preparing your new repository

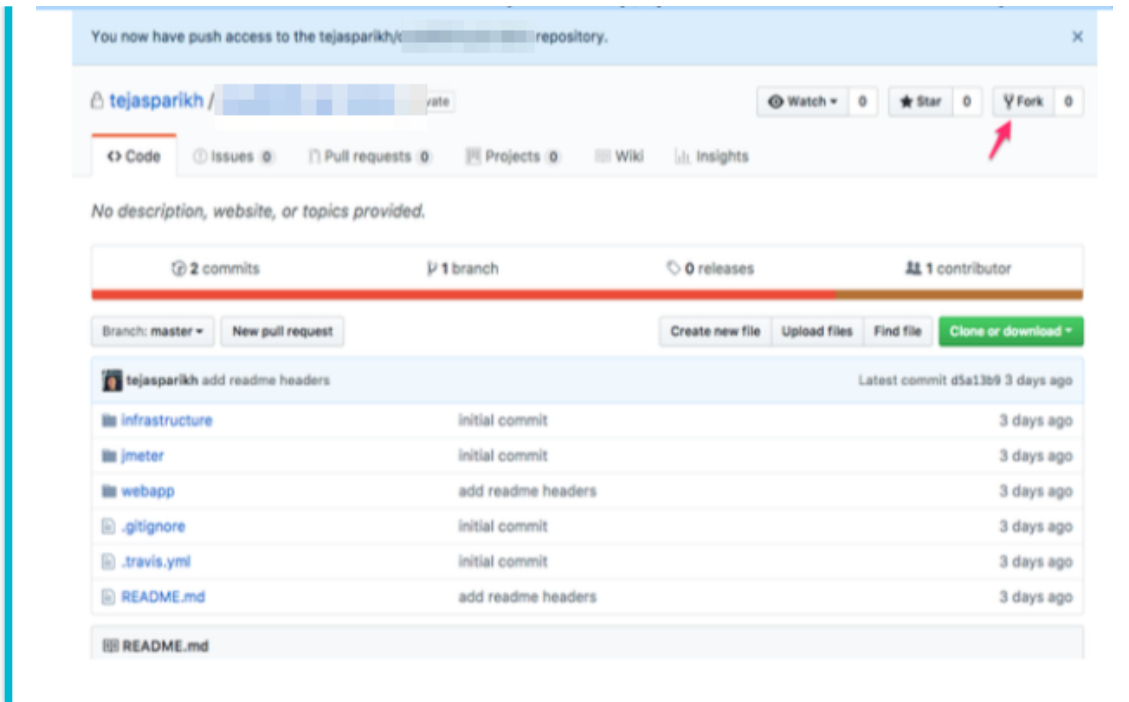
There is no need to keep this window open, we'll email you when the import is done.



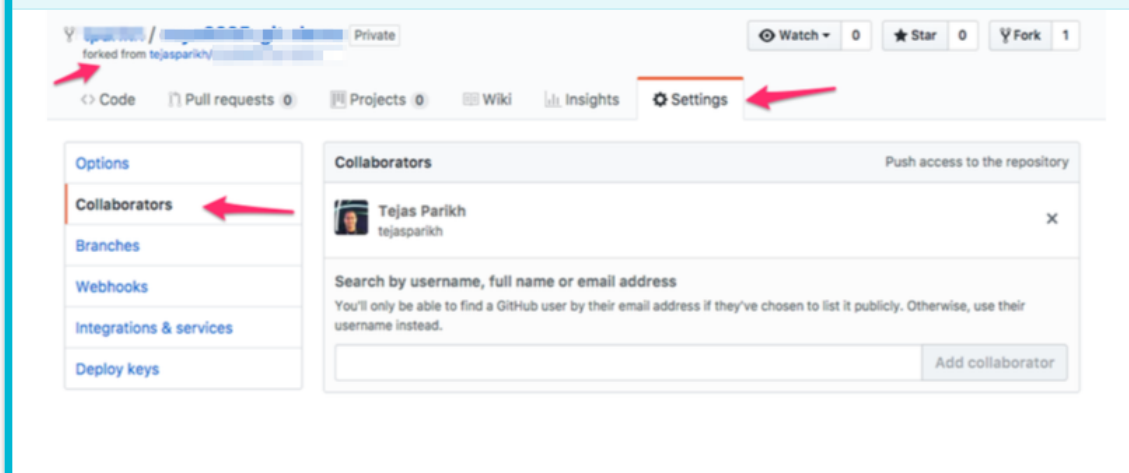
4.  Scrum Master should add other team members to their repository as collaborators 



5.  Once team members can see the repository, you should fork the repository. 



6. **Now each team members should add rest of the team members including scrum master to their repository as collaborators.**



7. Setup your local file system. I would recommend you create a course folder and a dev folder inside it by running the command `mkdir -p csye6225/app && mkdir -p csye6225/dev` in your terminal from your home directory.

8. **Change into `~/csye6225/dev` directory and clone your repository using the git clone command**

```

→ csye6225 git clone git@github.com:tejasparikh/csye6225-git-forking-workflow.git
Cloning into 'csye6225-git-forking-workflow'...
remote: Counting objects: 6, done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 6 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (6/6), 5.12 KiB | 874.00 KiB/s, done.
→ csye6225 ls -al
total 0
drwxr-xr-x  3 tejasparikh  wheel   96 Sep 12 20:18 .
drwxrwxrwt 20 root        wheel  640 Sep 12 20:16 ..
drwxr-xr-x  5 tejasparikh  wheel  160 Sep 12 20:18 csye6225-git-forking-workflow
→ csye6225 tree csye6225-git-forking-workflow
csye6225-git-forking-workflow
├── LICENSE
└── README.md

0 directories, 2 files
→ csye6225 █

```


9. Now add aliases to your team members repository using the command `git remote add <team_member_first_name> git@github.com:<TEAM_MEMBER_USERNAME>/csye6225-git-forking-workflow.git` Repeat the command for each one of your team members.
10. Fetch branches and/or from each one of your team members repository using the command `git fetch <team_member_first_name> .`
11. At this point you are ready to start working with your team on the newly created GitHub repository.

Working with Assignment Branches

All team members must execute following steps at the same time.






1. For each assignment you should create a new branch from the `master` branch using the command `git checkout -b <branch_name>`
2. Push this newly create branch to Github using the command `git push origin <branch_name> .`
3. Now start working on code changes. For the lab session make changes to any of the existing files and stage the changes using the `git add -A` command.
4. Commit the staged changes using the `git commit -m "COMMIT MESSAGE"` command.

5. Push the changes to the server using the command `git push origin <branch_name>`

6.  Now that everyone has pushed changes to their own `origin` repository, you should [create a pull request](https://help.github.com/articles/creating-a-pull-request/) [https://help.github.com/articles/creating-a-pull-request/] to your scrum master's repository.

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](#).

 base fork: `tejasparikh/csye6225-git-d...`  base: `assignment2`  + head fork: `tparikh/csye6225-git-demo`  compare: `assignment2` 

Scrum master's repo Assignment branch Your repository Your assignment branch

✓ Able to merge. These branches can be automatically merged.

 **Create pull request**

Discuss and review the changes in this comparison with others.

↔ 1 commit

📄 1 file changed

💬 0 commit comments

👤 1 contributor

📅 Commits on Jan 20, 2018

👤 tparikh



Update README.md

Verified 314922f

📄 Showing 1 changed file with 1 addition and 1 deletion.


Unified Split

2 README.md

  View 

... @@ -1,3 +1,3 @@

1 1 # CSYE 6225 Spring 2018 Starter Repository

7.  Review & merge pull request.

Update README.md #1 Edit

Open tparikh wants to merge 1 commit into `tejasparikh:assignment2` from `tparikh:assignment2`

Conversation 0 Commits 1 Files changed 1 +1 -1

tparikh commented just now
No description provided.

Update README.md Verified 314922f

Add more commits by pushing to the `assignment2` branch on `tparikh/csye6225-git-demo`.

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

Merge pull request Scrum master should merge the pull request
You can also open this in GitHub Desktop or view command line instructions.

Write Preview AA B i “ > ↵ ⋮ ⋮ ⋮ ↶ @

Merged tparikh merged 1 commit into `tejasparikh:assignment2` from `tparikh:assignment2` just now

Conversation 0 Commits 1 Files changed 1

tparikh commented a minute ago
No description provided.

Update README.md Verified 314922f

tparikh merged commit `8fb0aab` into `tejasparikh:assignment2` just now Revert

✓ Pull request successfully merged and closed
You're all set—the `tparikh:assignment2` branch can be safely deleted.
Delete branch Do not delete the branch

Write Preview AA B i “ > ↵ ⋮ ⋮ ⋮ ↶ @

Keeping Your Fork in Sync

1. Now that the code is merged sync your code with team member's repository. To pull latest changes from scrum master's repository, run the command `git pull <team_member_first_name> <branch_name>` in your repository.

2. Push the changes to your repository using the command `git push origin <branch_name>` . There are no changes to stage and nothing to commit in this case.

