Git Help

An introduction to Git and GitHub

What is Git?

Why use Git?

Keeping track of code changes

- `git init`
- `git add`
- `git commit`
- `git log`
- `git status`

git init

- Create a new git repository

```
Ω ~/git_example/ git init
Initialized empty Git repository in /home/chad/git_example/.git/
```

git add

- Tell git that a newly created/changed file should be included in the next “commit” (snapshot)
- Sometimes called “staging” a change

```
Ω ~/git_example/ git add hello.py
```

git commit

- Save a snapshot of the current version of the code
- Include a human-readable message

```
Ω ~/git_example/ git commit -m "Created hello.py"
[master (root-commit) 5c8c6ff] Created hello.py
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 hello.py
```

git commit

- Save a snapshot of the current version of the code
- Include a human-readable message

```
Ω ~/git_example/ git commit -m "Created hello.py"
[master (root-commit) 5c8c6ff] Created hello.py
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 hello.py
```

git log

- View the commit history

```
Ω ~/git_example/ git log --oneline
5c8c6ff (HEAD -> master) Created hello.py
```

git log

```
Ω ~/Documents/cs50/check50/ git log --oneline
f942cce (HEAD -> v3, origin/v3) update compiles method
baebff6 recurse submodules when fetching
5eb7e52 add update features
0588801 fix check50 case
6e70904 updated excepted branches and tags regex
61b8c21 Fix branch in travis badge
ac7be31 fix capitalization in index
df1cbf3 add check50.py to docs
611c1cb tweak flask exception message
bc08969 enable --local by defaualt
be406d4 slightly improve docs
4b26d2b renamed process + updated docs
cf29235 improve docs + remove diff
```

git status

- View the current state of the repository
- Show modified, staged, and untracked files

```
Ω ~/git_example/ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   hello.py

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        goodbye.py

no changes added to commit (use "git add" and/or "git commit -a")
```

git status

- View the current state of the repository
- Show modified, staged, and untracked files

```
Ω ~/git_example/ git add hello.py
Ω ~/git_example/ git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        modified:   hello.py

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        goodbye.py
```

Now, you try!

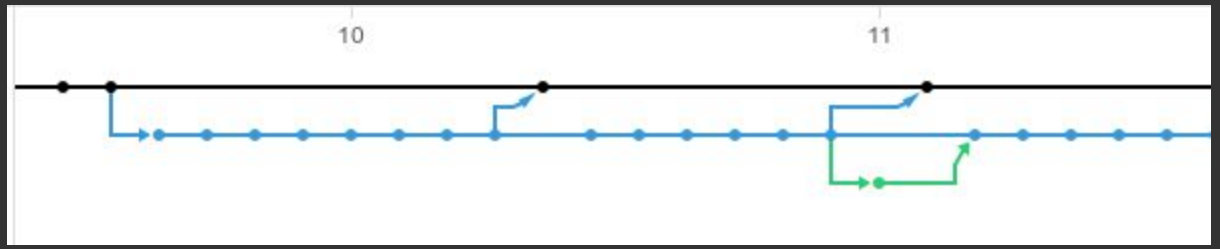
1. Open up a terminal in your Project 1 folder
2. Initialize a git repository
3. Add the files you want to commit
4. Create your first commit (don’t forget to specify a message with `-m`)
5. View the log

If curious, run `git status` between each of the above steps.

Keeping multiple versions of your code

- `git branch`
- `git checkout`
- `git merge`

Git branches



https://github.com/cs50/style50/network

git branch

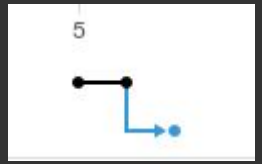
- List, create, and delete branches

```
Ω ~/git_example/ git branch
* master
```

git branch

- List, create, and delete branches

```
Ω ~/git_example/ git branch name
Ω ~/git_example/ git branch
* master
  name
```



git checkout

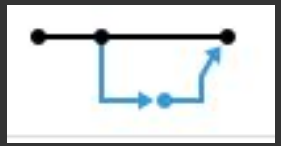
- Switch to another branch

```
Ω ~/git_example/ git branch
* master
  name
Ω ~/git_example/ git checkout name
Switched to branch 'name'
Ω ~/git_example/ git branch
  master
* name
```

git merge

- Join two branches together, merging their commit histories

```
Ω ~/git_example/ git merge name
Updating bbce4eb..d94257e
Fast-forward
 hello.py | 3 ++-
 1 file changed, 2 insertions(+), 1 deletion(-)
```



Reverting changes

- `git reset`
- `git checkout (continued)`

git reset

- Returns code to a previous state
- Careful, cannot be undone!

```
Ω ~/git_example/ cat hello.py
name = input("Enter name: ")
print(f"hello, {name}")
Ω ~/git_example/ git log --oneline
d94257e (HEAD -> master, name) ask for name
bbce4eb added print statement
5c8c6ff Created hello.py
Ω ~/git_example/ git reset --hard bbce4eb
HEAD is now at bbce4eb added print statement
Ω ~/git_example/ cat hello.py
print("hello, world!")
Ω ~/git_example/ git log --oneline
bbce4eb (HEAD -> master) added print statement
5c8c6ff Created hello.py
```

`git checkout` (continued)

- View the state of the repository at a particular commit

```
Ω ~/git_example/ git checkout bbce4eb
Note: checking out 'bbce4eb'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by performing another checkout.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -b with the checkout command again. Example:

  git checkout -b <new-branch-name>

HEAD is now at bbce4eb added print statement
```

`git checkout` (continued)

To review:

```
$ git checkout <BRANCH_NAME>
```

switches to a different branch

```
$ git checkout <COMMIT_HASH>
```

allows you a (read-only) view of an earlier commit

Now, you try!

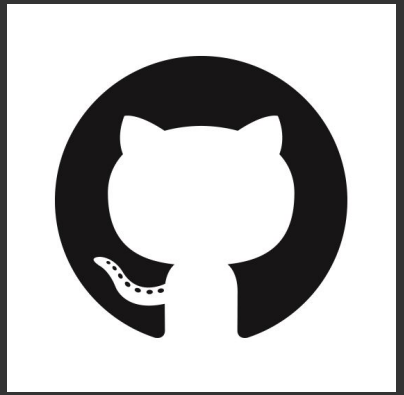
1. Create a new branch in your Project 1 repository
2. Make a small change (add a comment)
3. Merge this change back into master
4. Use `git reset` to revert this change

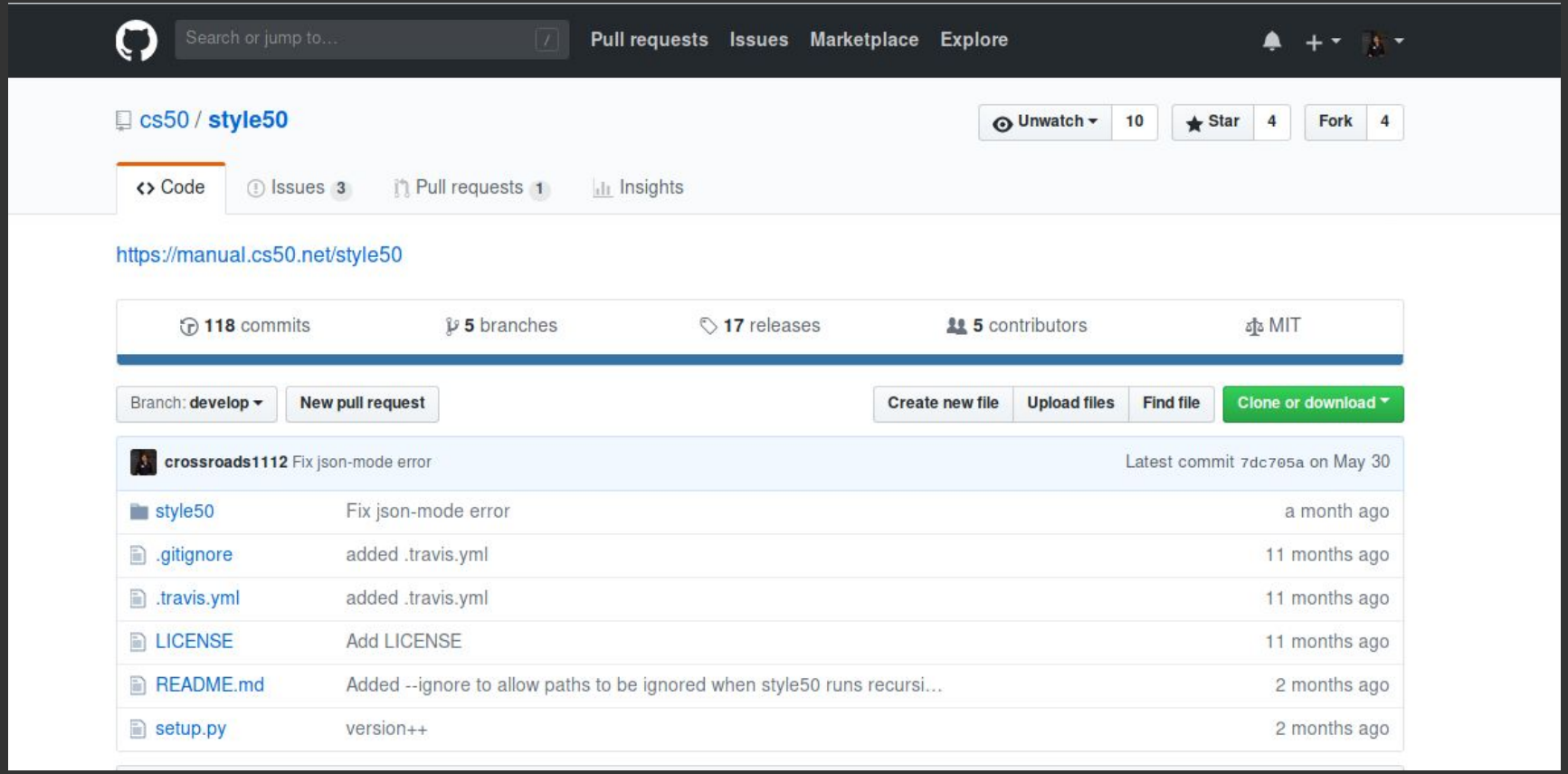
Collaborating with others

- `git remote`
- `git push`
- `git pull`

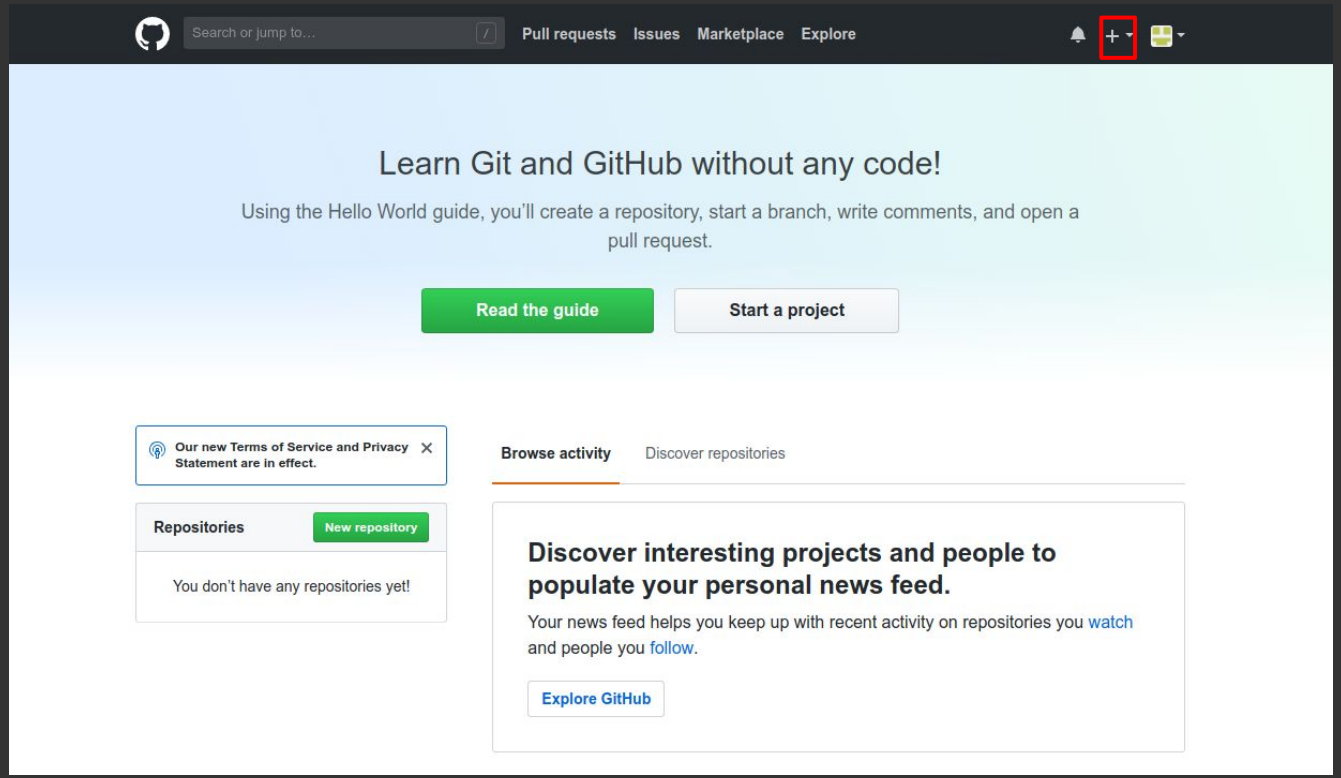
But first...

GitHub

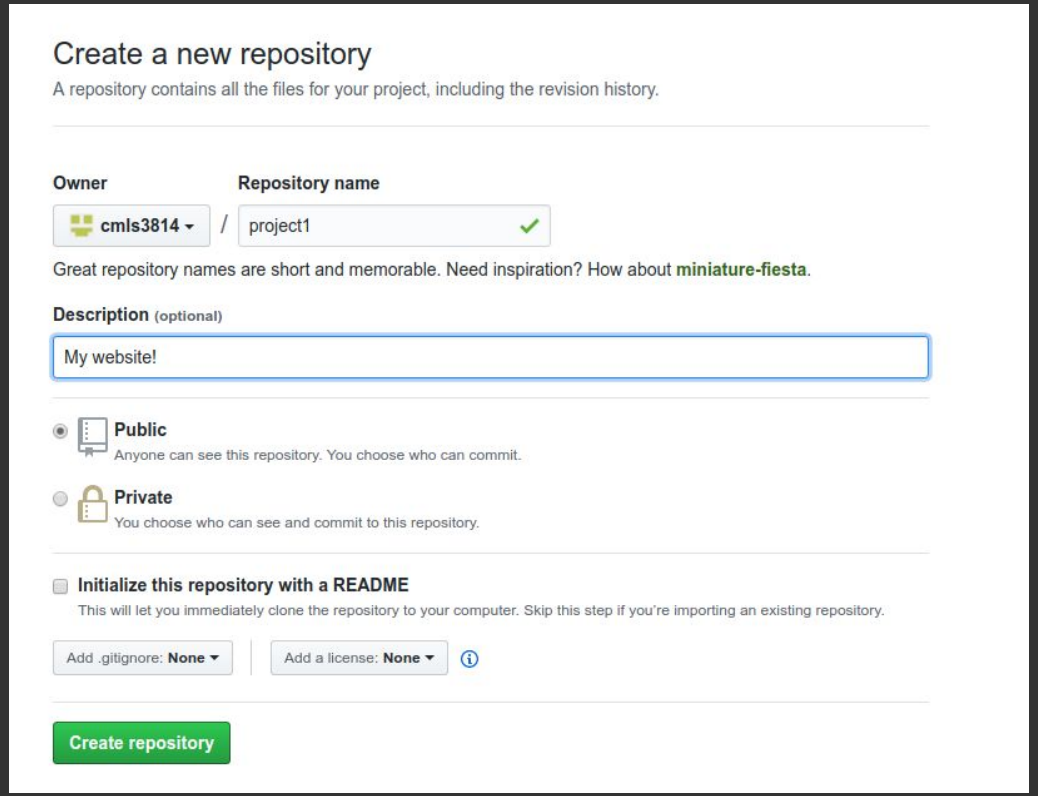



Search or jump to...
Pull requests
Issues
Marketplace
Explore
cs50 / style50
Unwatch 10
Star 4
Fork 4
Code
Issues 3
Pull requests 1
Insights
https://manual.cs50.net/style50
118 commits
5 branches
17 releases
5 contributors
MIT
Branch: develop
New pull request
Create new file
Upload files
Find file
Clone or download
crossroads1112 Fix json-mode error
Latest commit 7dc705a on May 30
style50 Fix json-mode error a month ago
.gitignore added .travis.yml 11 months ago
.travis.yml added .travis.yml 11 months ago
LICENSE Add LICENSE 11 months ago
README.md Added --ignore to allow paths to be ignored when style50 runs recursi... 2 months ago
setup.py version++ 2 months ago

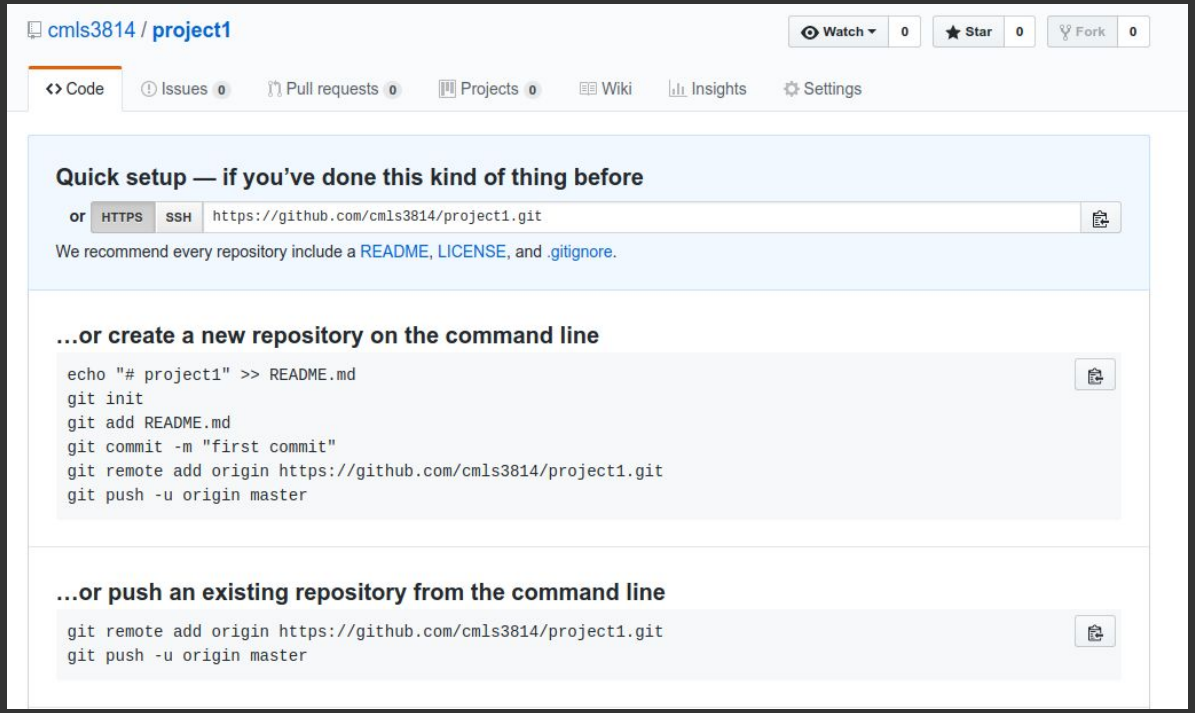
Create a new repository



Create a new repository



Create a new repository



git remote

- Manage remote repositories

```
Ω ~/project1/ git remote add origin https://github.com/cmls3814/project1.git
Ω ~/project1/ git remote show origin
* remote origin
  Fetch URL: https://github.com/cmls3814/project1.git
  Push  URL: https://github.com/cmls3814/project1.git
  HEAD branch: (unknown)
```

git push

- Update remote repository with any commits made locally

```
Ω ~/project1/ git push -u origin master
Username for 'https://github.com': cmls3814
Password for 'https://cmls3814@github.com':
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 286 bytes | 286.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/cmls3814/project1.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

git pull

- Update local repository with any changes made remotely

```
Ω ~/project1/ git pull
remote: Counting objects: 3, done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/cmls3814/project1
   1d4e126..e5d22b1  master     -> origin/master
Updating 1d4e126..e5d22b1
Fast-forward
 index.html | 3 +--
 1 file changed, 1 insertion(+), 2 deletions(-)
```

Now, you try!

- Create your Project 1 repository
- Add https://github.com/<YOUR_USERNAME>/project1.git as a remote repository
- Push your website to the remote repository

Congrats! You've created your first open-source project.

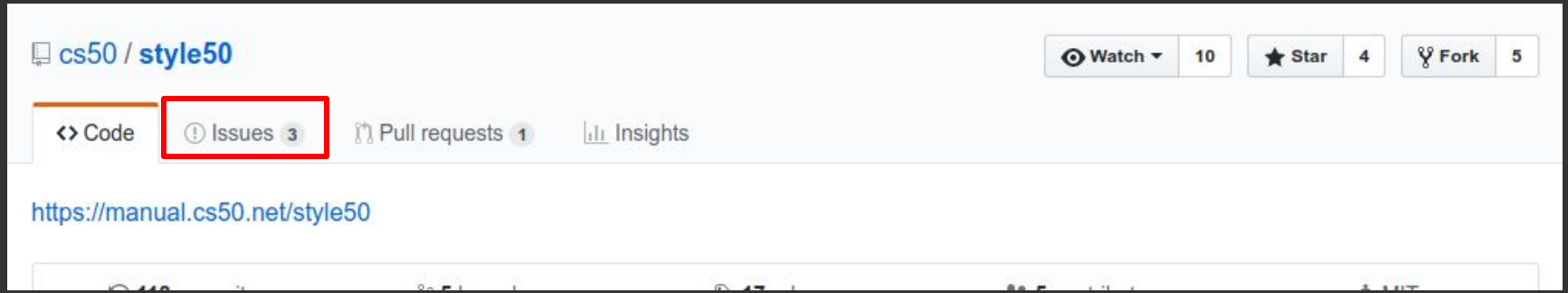
Oh no, merge conflicts!

Contributing to other repositories

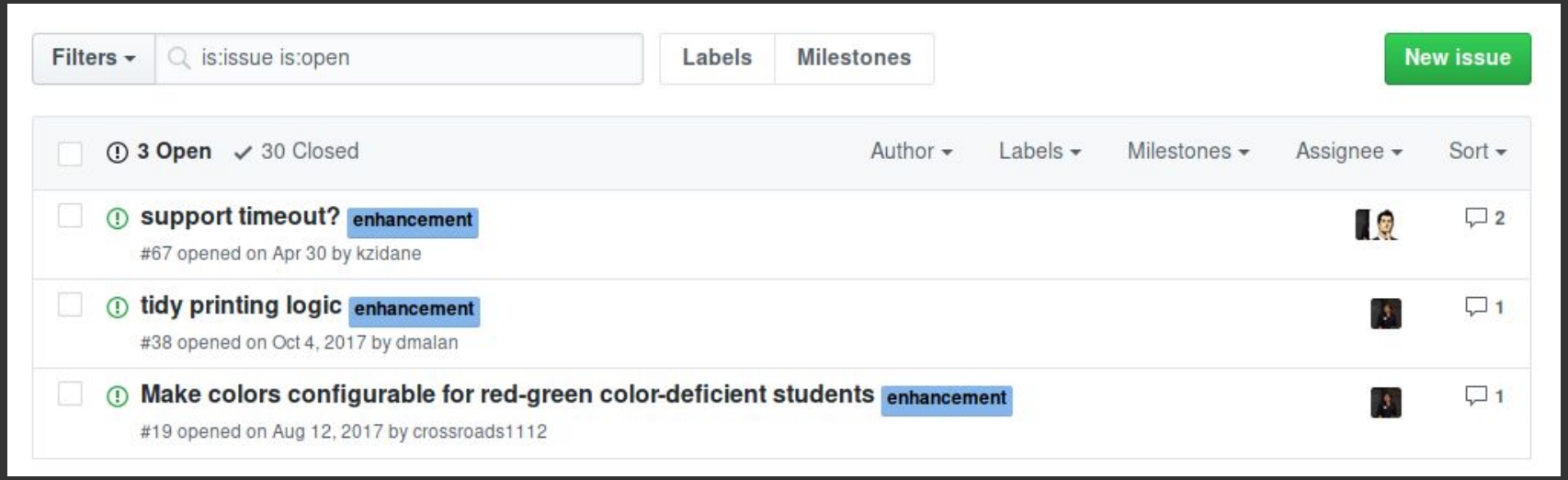
- Filing issues on Github
- Forking a repository on Github
- `git clone`
- Creating pull requests

Issues

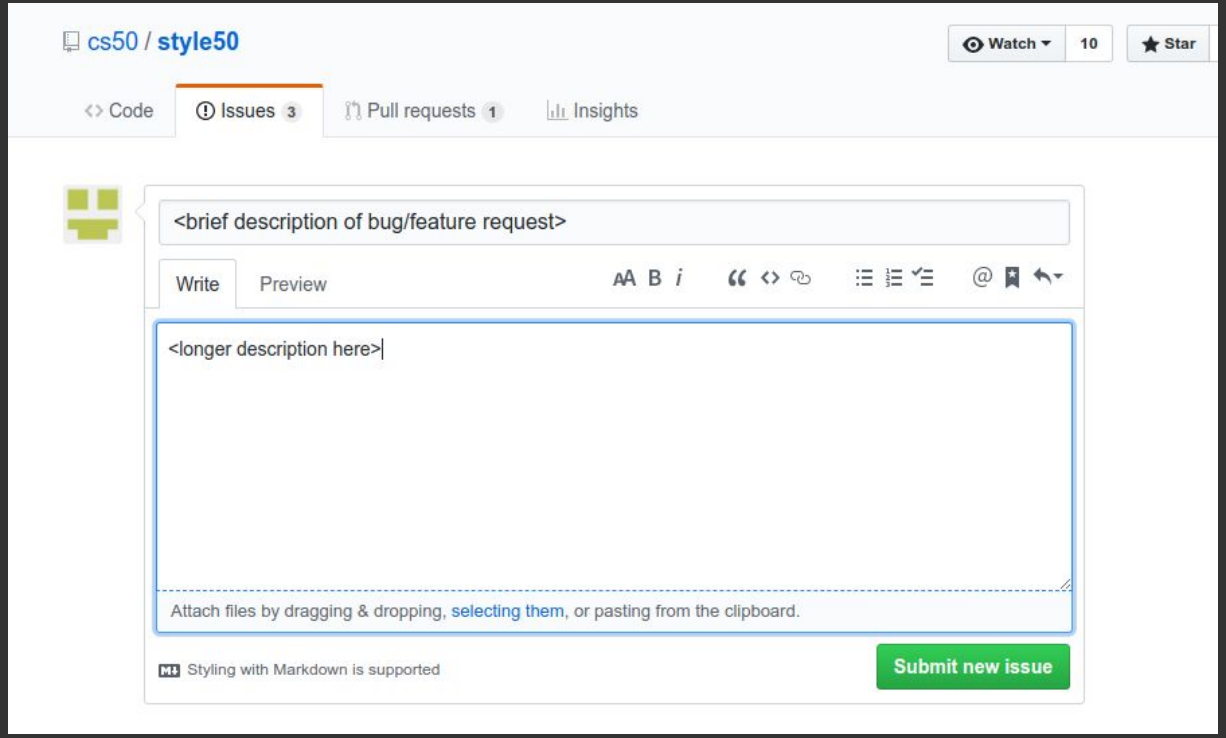
- Bug reports/feature requests



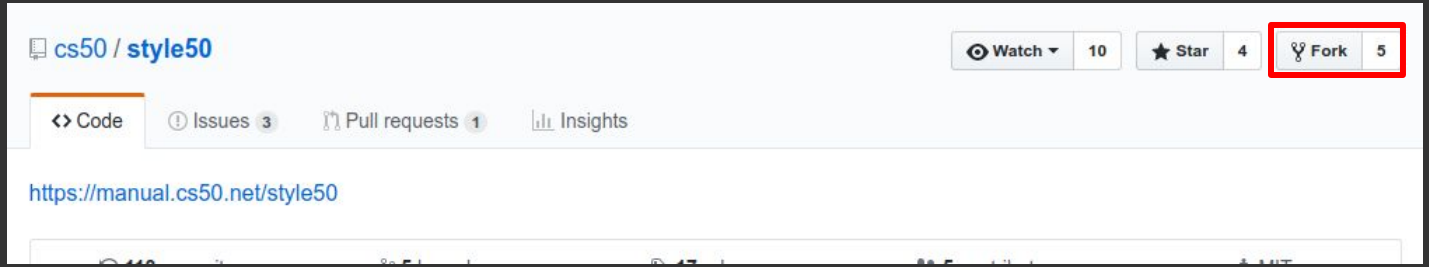
Filing issues on GitHub



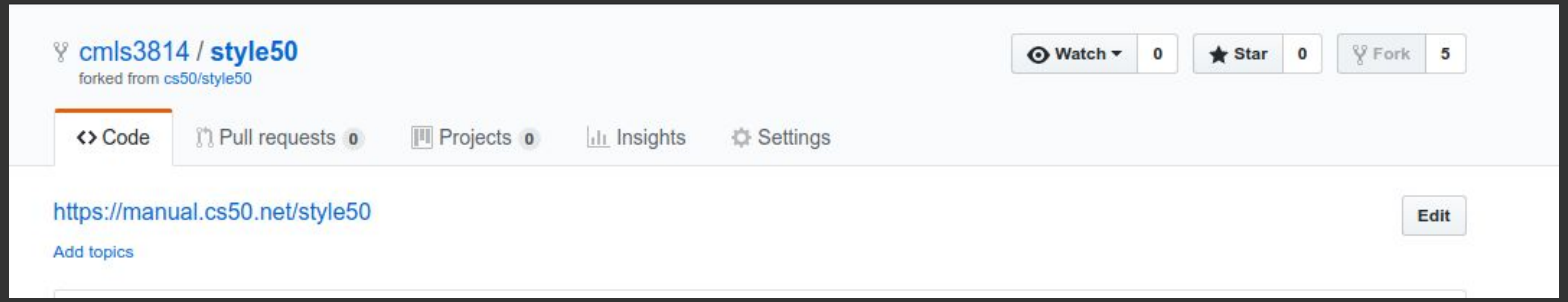
Filing issues on GitHub



Forking a repository on Github



Forking a repository on Github



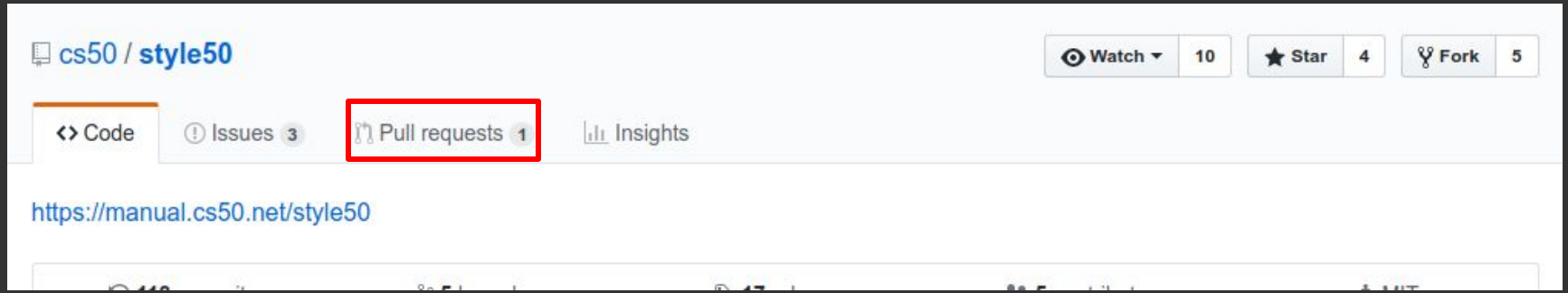
git clone

- Copy an existing remote repository locally

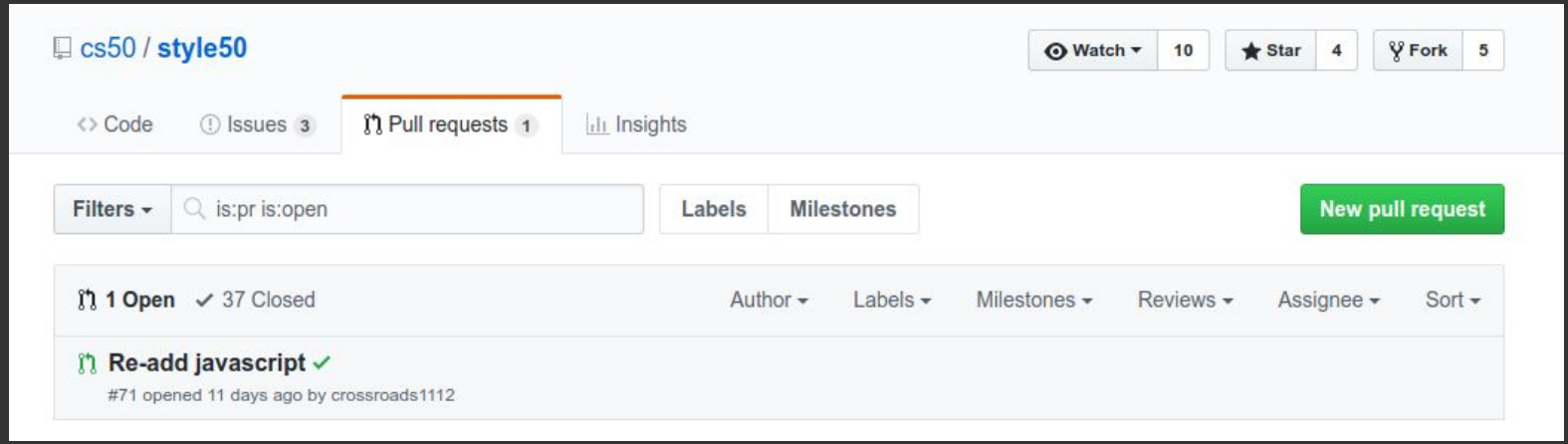
```
Ω ~/ git clone https://github.com/cmls3814/style50.git
Cloning into 'style50'...
remote: Counting objects: 495, done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 495 (delta 4), reused 9 (delta 4), pack-reused 481
Receiving objects: 100% (495/495), 86.39 KiB | 2.33 MiB/s, done.
Resolving deltas: 100% (306/306), done.
```

Pull Requests

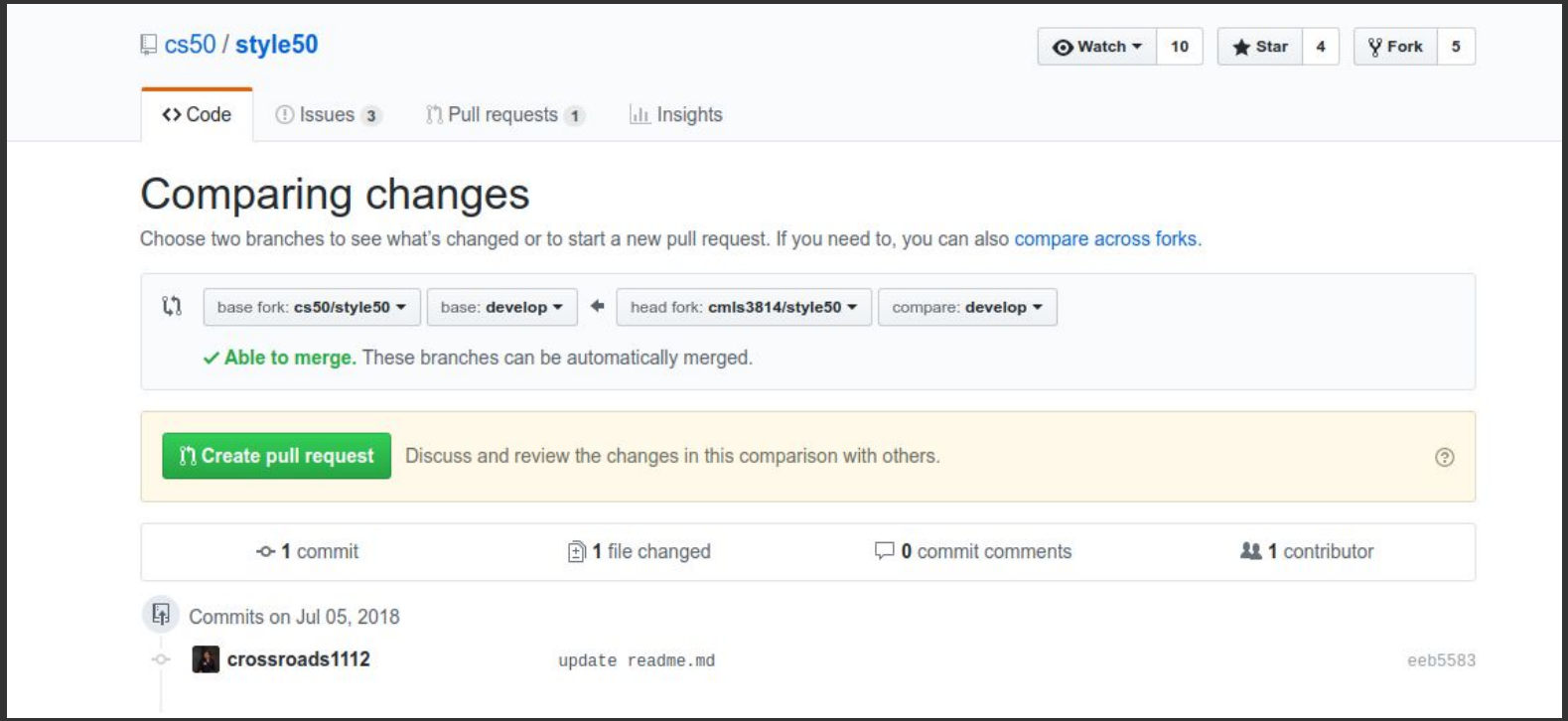
- Request that the original repository adopt the changes you made



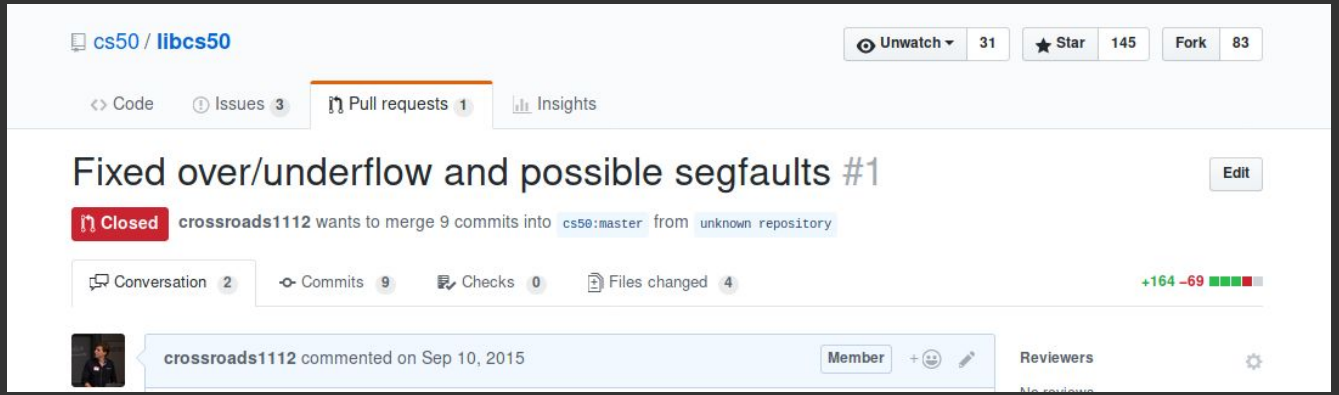
Creating pull requests



Creating pull requests



Why contribute to other projects?


cs50 / libcs50
Unwatch 31
Star 145
Fork 83
Code
Issues 3
Pull requests 1
Insights
Fixed over/underflow and possible segfaults #1
Edit
Closed
crossroads1112 wants to merge 9 commits into cs50:master from unknown repository
Conversation 2
Commits 9
Checks 0
Files changed 4
+164 −69
crossroads1112 commented on Sep 10, 2015
Member
Reviewers

Now, you try!

1. Find a partner
2. Exchange links to your respective repositories
3. Fork and clone their repository into the IDE and try to improve it some way.
4. Push your changes to your fork, and create a pull request on their repository.

tinyurl.com/feedback14300

lunch!