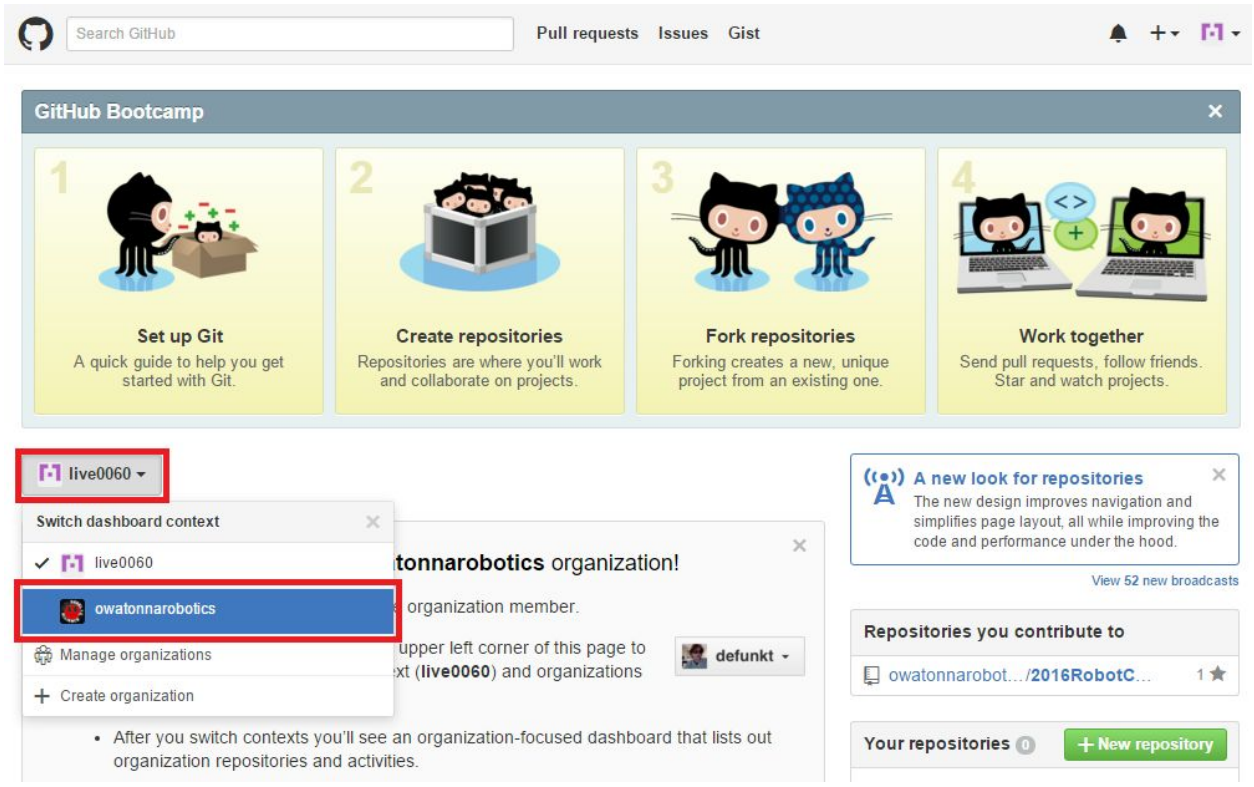
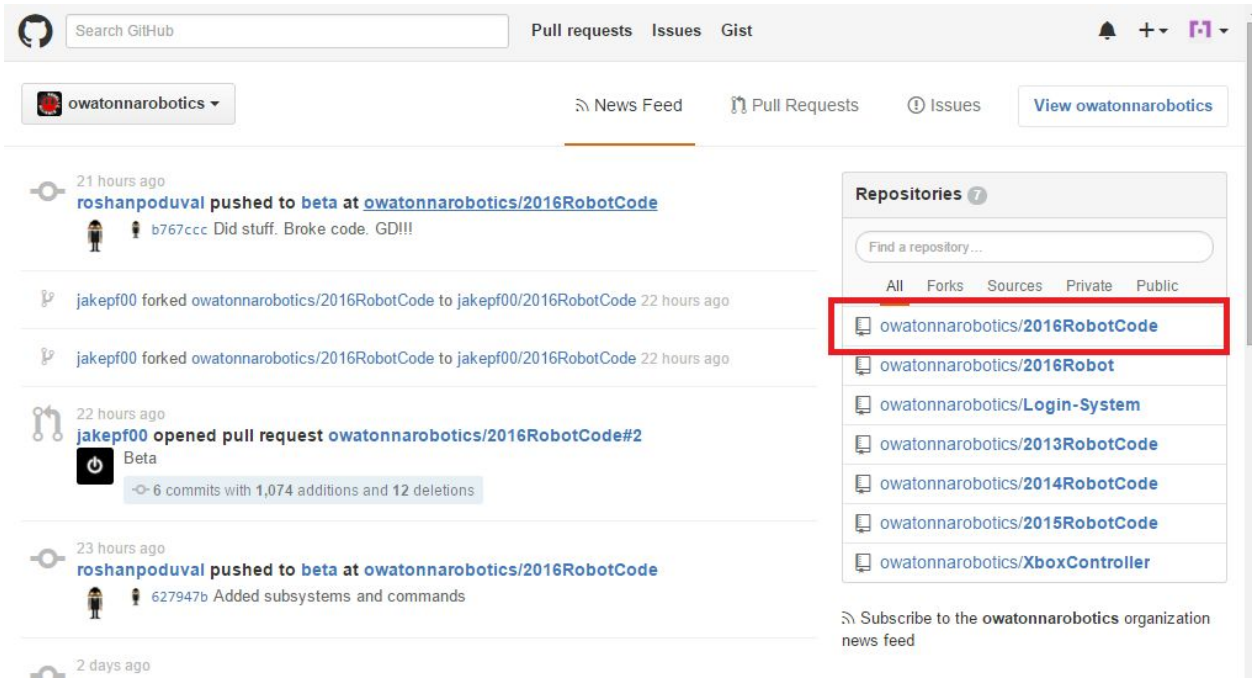
Github Project Instructions

Step 1: Obtain a fork of the original Github project.

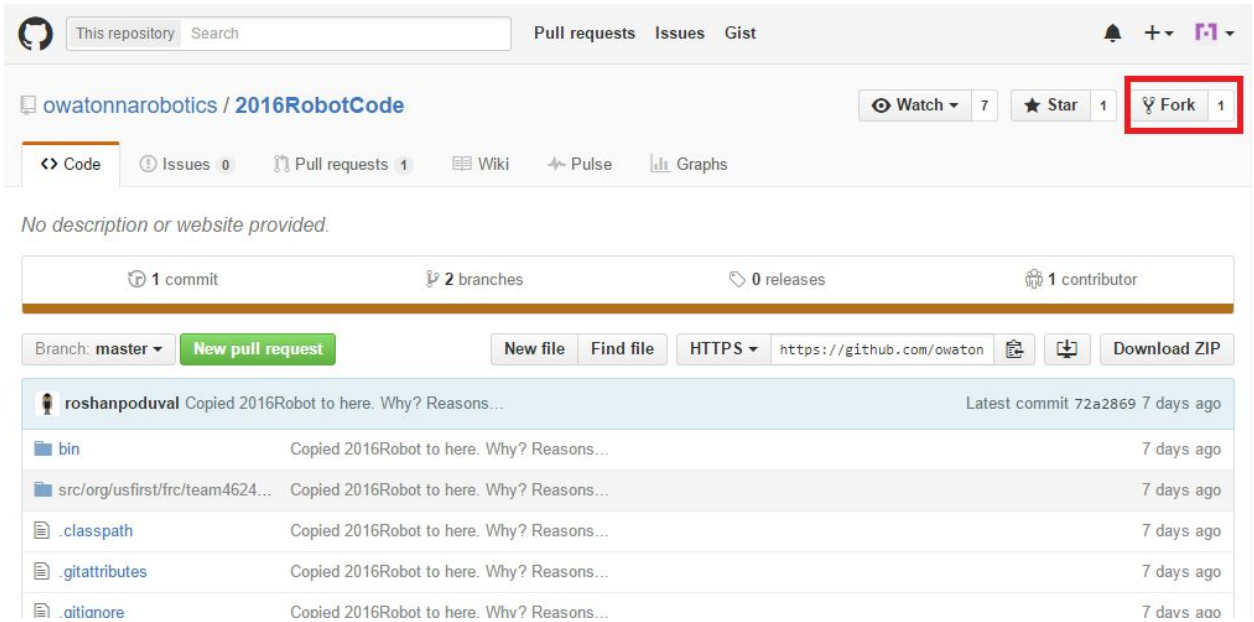
Log in to Github.com and navigate to the owatonnaroboitcs dashboard.



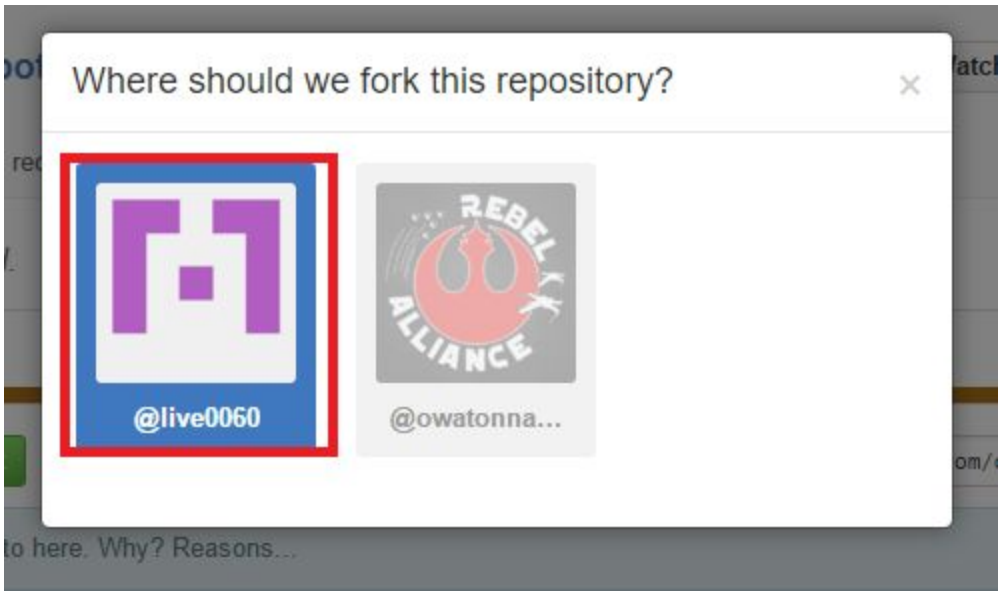
Navigate to the 2016RobotCode repository.



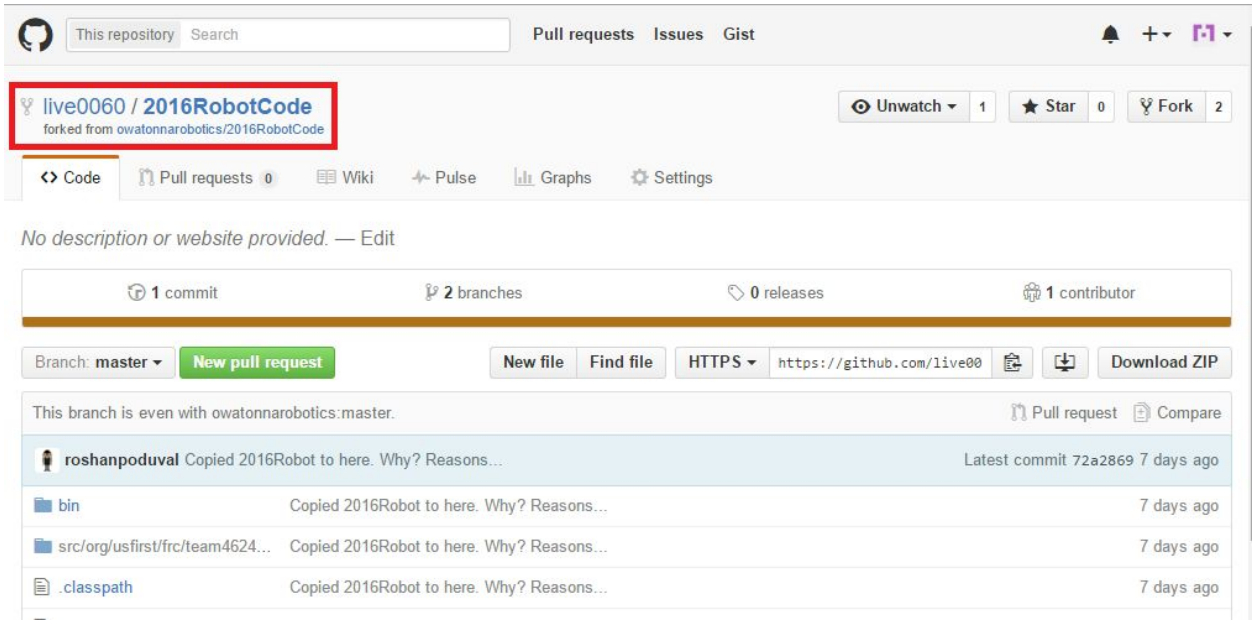
Fork the repository to your Github account.



Select your account.

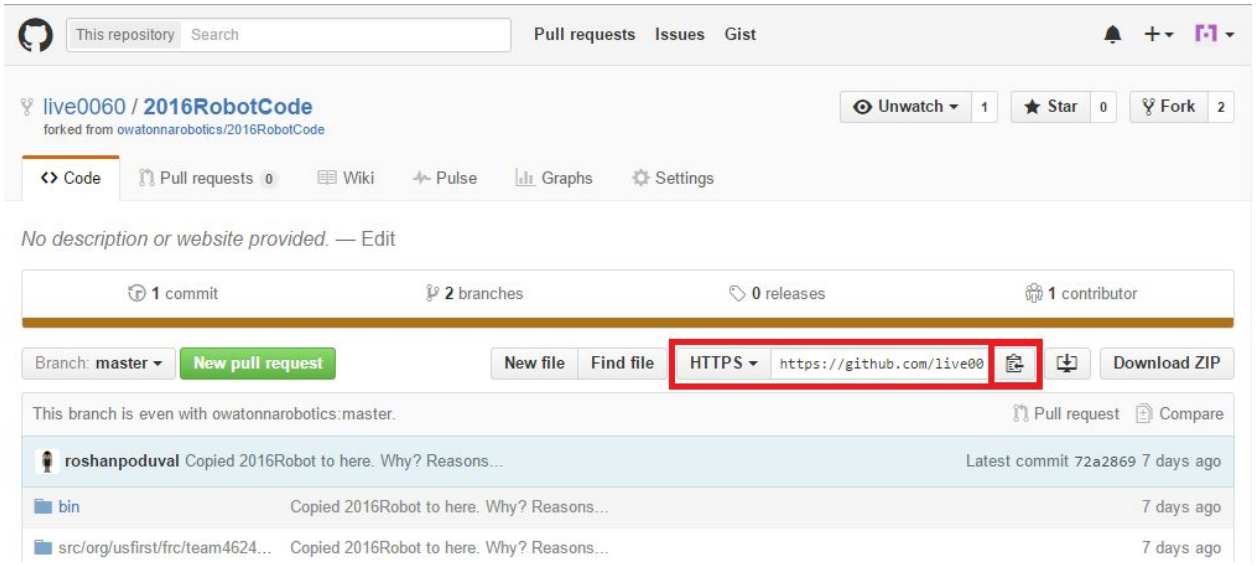


Profit!!!


This repository
Search
Pull requests
Issues
Gist
live0060 / 2016RobotCode
forked from owatonnarobotics/2016RobotCode
Unwatch 1
Star 0
Fork 2
Code
Pull requests 0
Wiki
Pulse
Graphs
Settings
No description or website provided. — Edit
1 commit
2 branches
0 releases
1 contributor
Branch: master
New pull request
New file
Find file
HTTPS
https://github.com/live00
Download ZIP
This branch is even with owatonnarobotics:master.
Pull request
Compare
roshanpoduval Copied 2016Robot to here. Why? Reasons...
Latest commit 72a2869 7 days ago
bin
Copied 2016Robot to here. Why? Reasons...
7 days ago
src/org/usfirst/frc/team4624...
Copied 2016Robot to here. Why? Reasons...
7 days ago
.classpath
Copied 2016Robot to here. Why? Reasons...
7 days ago

Step 2: Clone the fork onto your computer.

Copy the clone's URL.



Open Git Shell. (CMD, Bash, Powershell, etc.)

Enter the following command: git clone <URL>

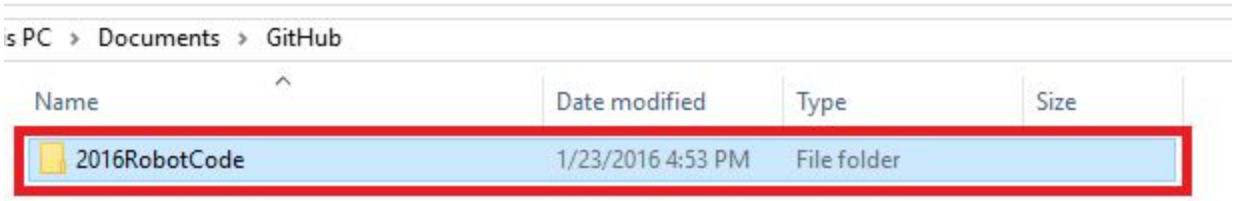
```
C:\Users\Bolognal\Documents\GitHub> git clone https://github.com/live0060/2016RobotCode.git
Cloning into '2016RobotCode'...
remote: Counting objects: 214, done.
remote: Compressing objects: 100% (114/114), done.
remote: Total 214 (delta 66), reused 183 (delta 35), pack-reused 0
,
Receiving objects: 100% (214/214), 3.11 MiB | 1.75 MiB/s, done.
Resolving deltas: 100% (66/66), done.
Checking connectivity... done.
```

Navigate to the project folder. Enter the following command: cd 2016RobotCode
Hint: TAB to autocomplete.

Switch to branch “beta”. Enter the following command: git checkout beta

```
C:\Users\Bolognal\Documents\GitHub> cd .\2016RobotCode
C:\Users\Bolognal\Documents\GitHub\2016RobotCode [master]> git checkout beta
Switched to branch 'beta'
C:\Users\Bolognal\Documents\GitHub\2016RobotCode [beta]>
```

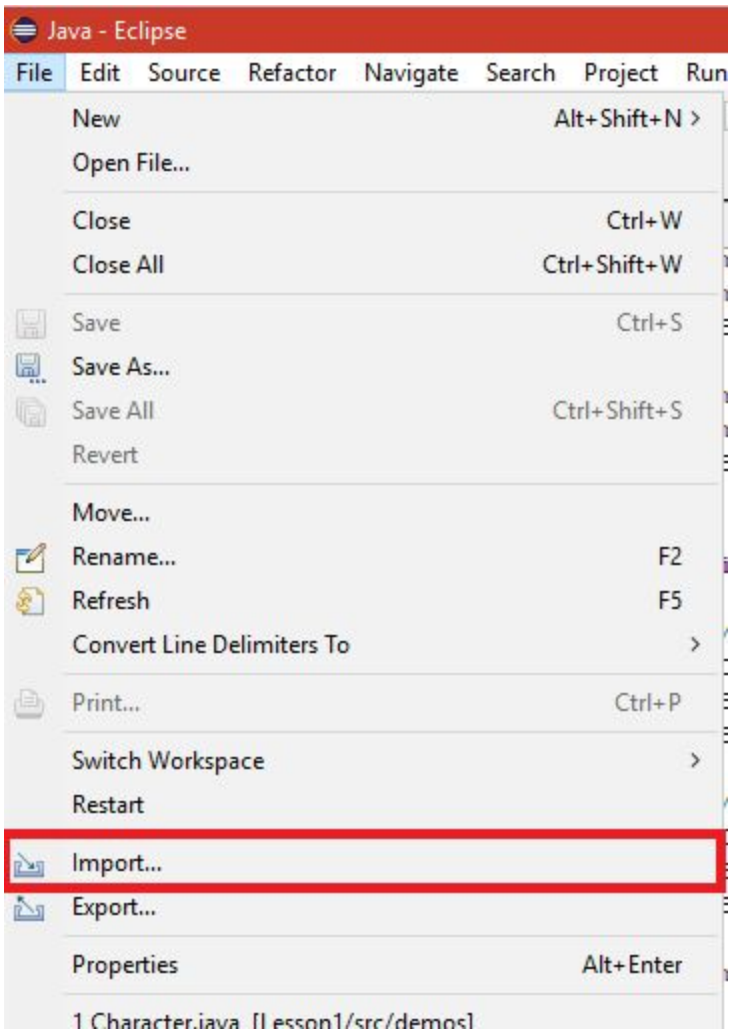
Profit!!!



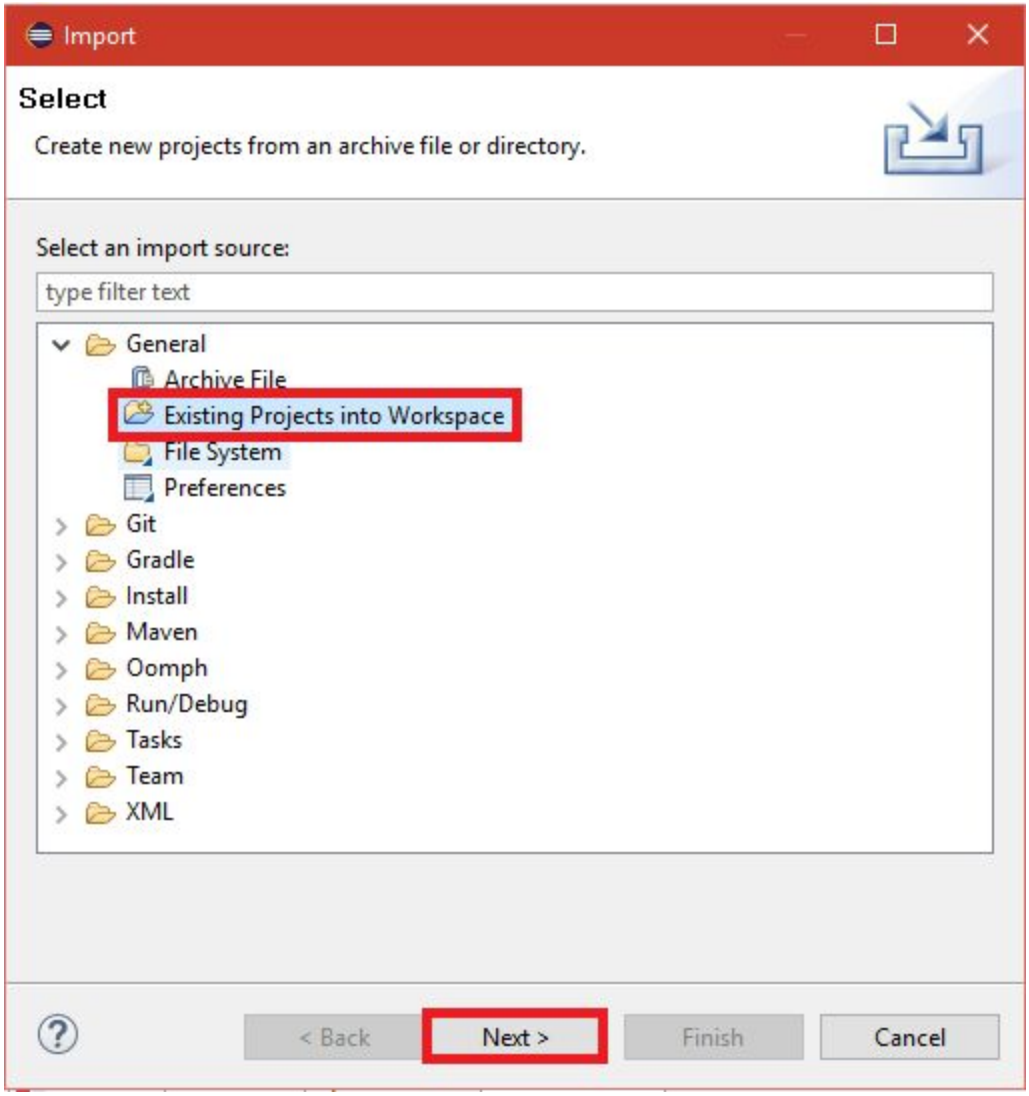
Step 3: Open the project in Eclipse.

Open Eclipse.

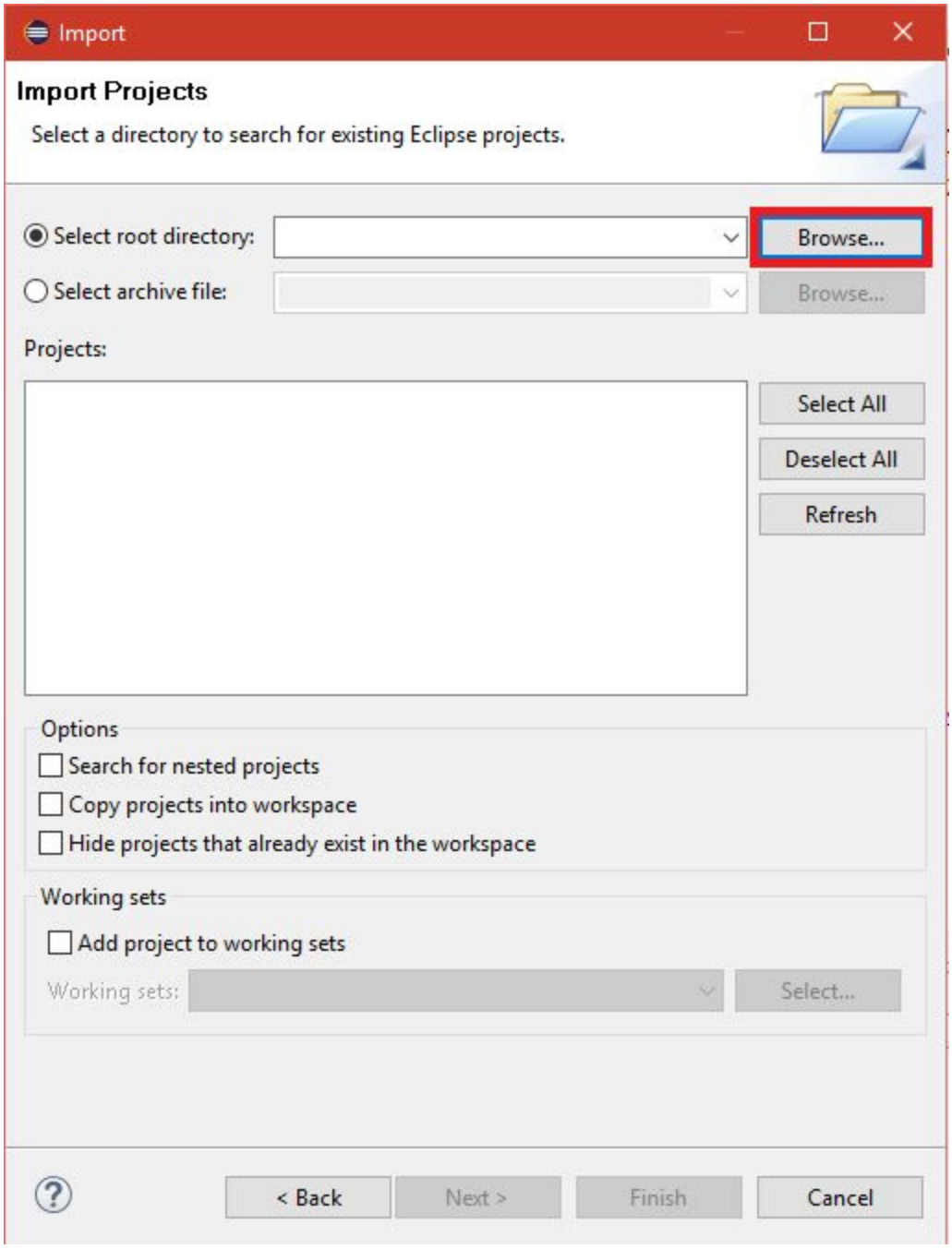
Import the project.



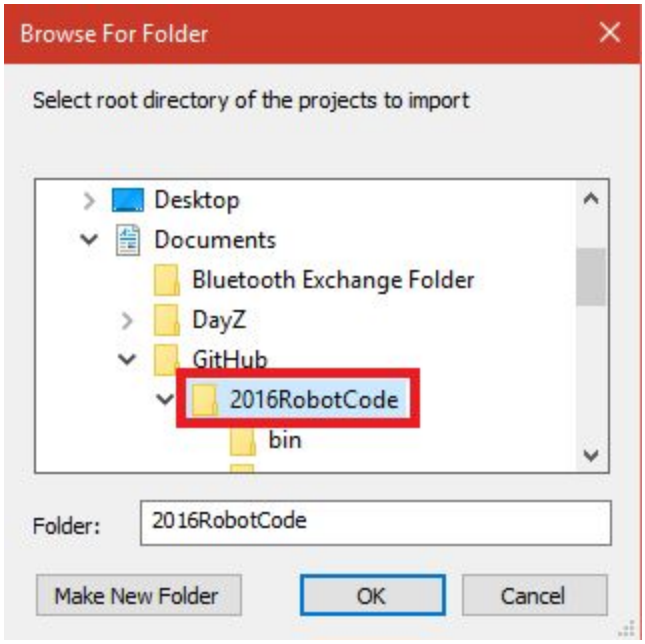
Import an “Existing Project into Workspace”.



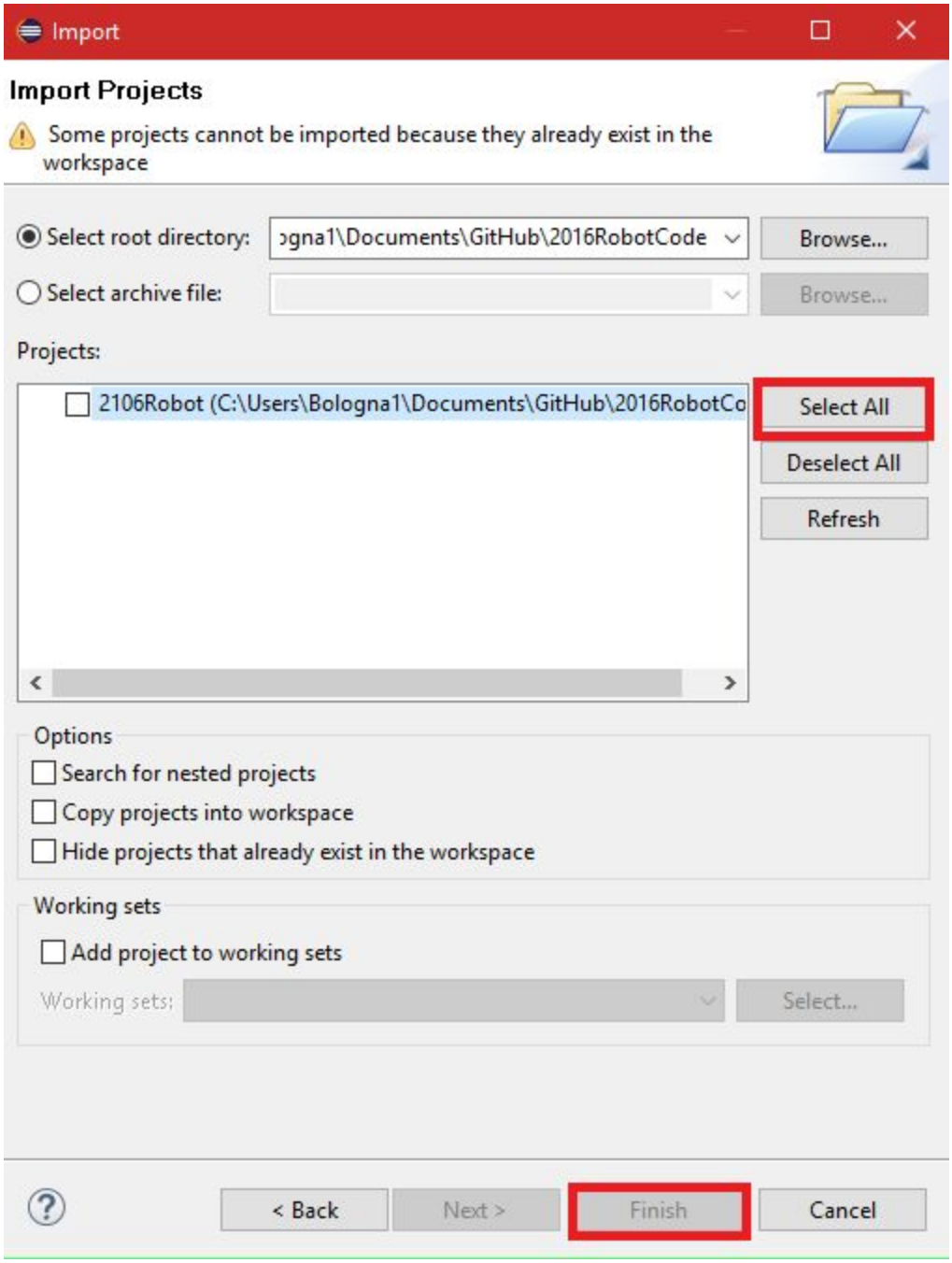
Browse for the project folder.



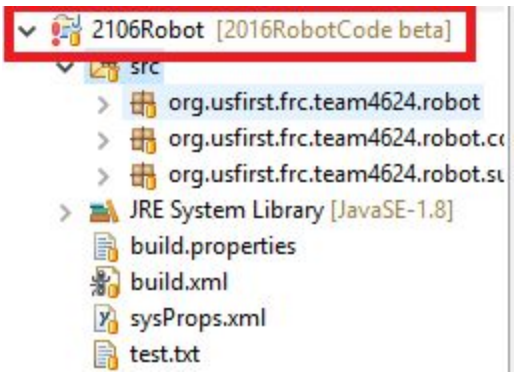
Select the project folder from your Github folder.



“Select All” and “Finish”.



Profit!!!



Step 4: Make your changes to the project.

<u>OPTIONAL</u> STEPS TO SYNC A FORK WITH ORIGINAL PROJECT:

FIRST TIME SETUP:

Step 1: In Git Shell, navigate to the project folder.

Copy the ORIGINAL PROJECT URL. We want to sync with it.

Step 2: Enter the following command: git remote add upstream <URL>

Note: The “git remote -v” commands show the remote repositories for your fork.

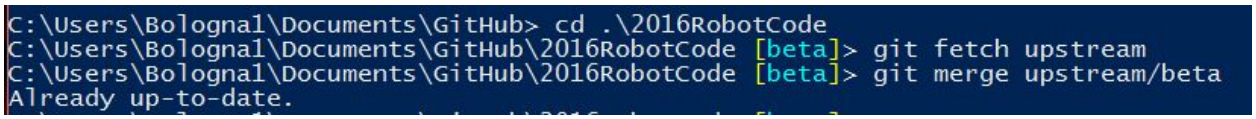
Step 1: In the project folder, enter the following command: git fetch upstream

This puts the latest version of the original code in the upstream repository.

Step 2: Switch to the branch “beta”.

Step 3: Enter the following command: git merge upstream/beta

This syncs your fork up with the latest version of the original project.



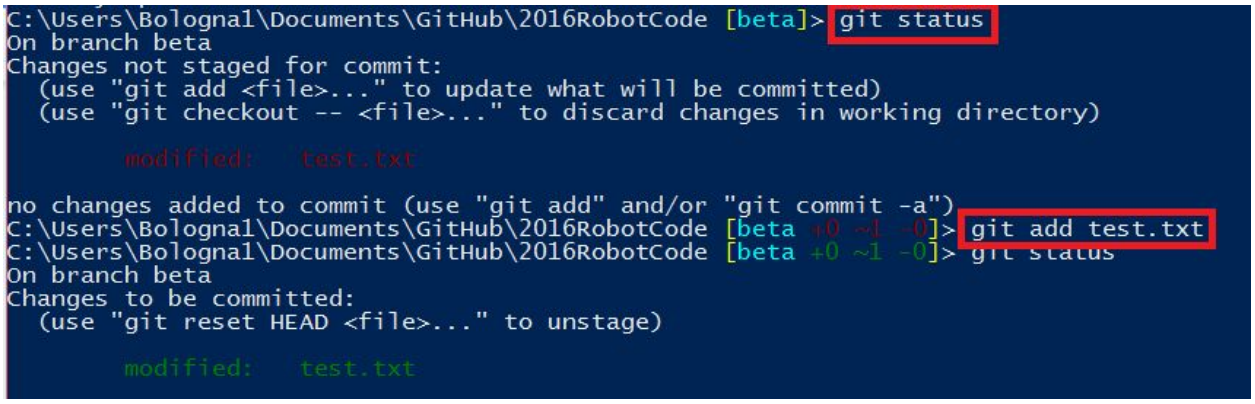
Profit!!!

Step 5: Committing your changes.

Open the Git Shell and navigate to the project folder.
(Make sure you're on branch "beta".)

Pro tip: Check to see what changes have not been added to the commit yet. (RED)
Enter the following command: git status

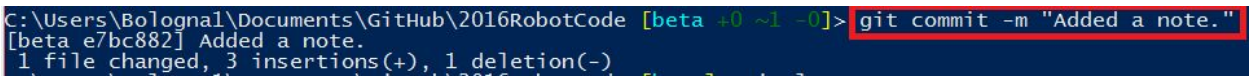
Enter the following command to add a file to the commit: git add <FILENAME>



Notice that "git status" shows "test.txt" under "Changes to be committed:" in GREEN after it got added.

Once you've added all the changes you're going to commit, enter the following command: git commt -m "<MESSAGE>"

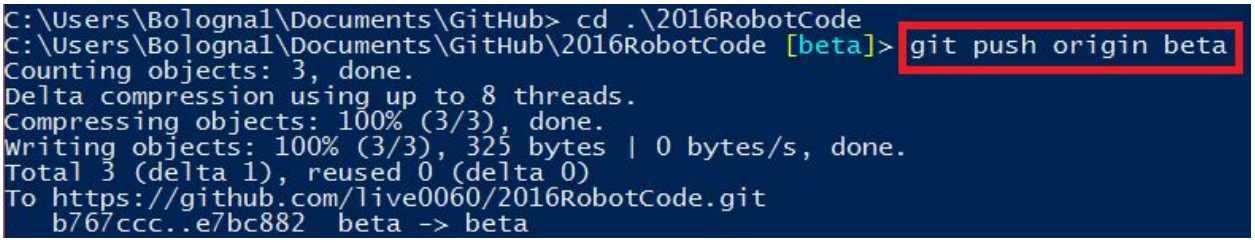
Always give the commit a short message telling everyone what you changed.



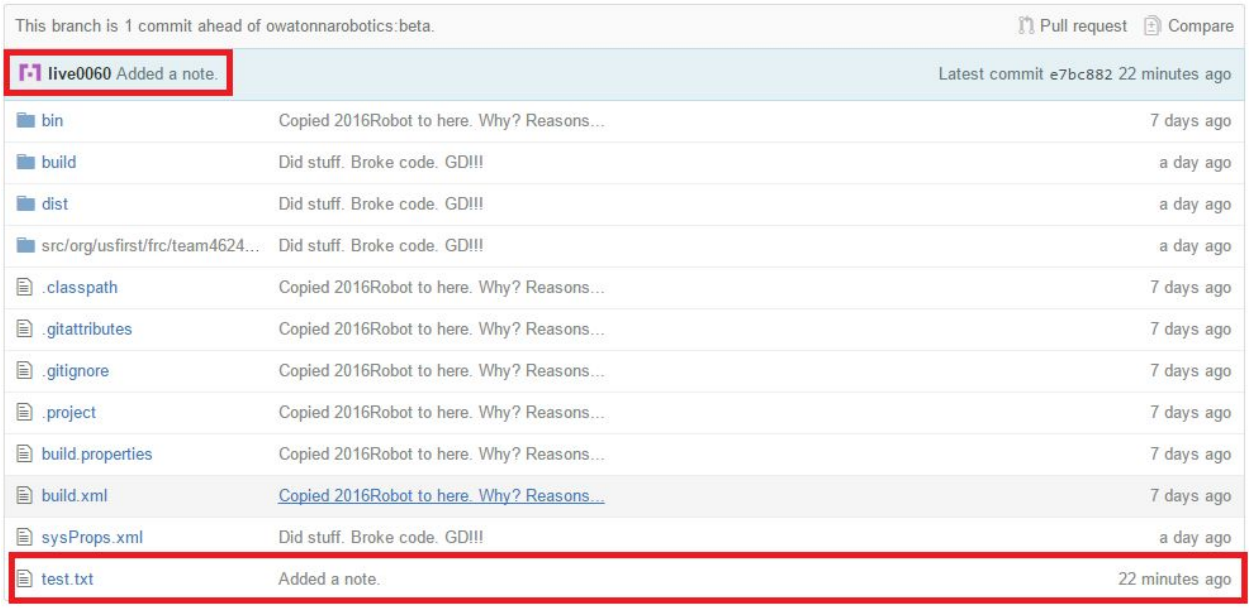
Profit!!!

Step 6: Merging your fork with the original project.

After you've finished making changes to the project and have committed them to your local repository, you have to push the branch to your remote repository. Using the Git Shell, enter the following command in your project folder: git push origin beta

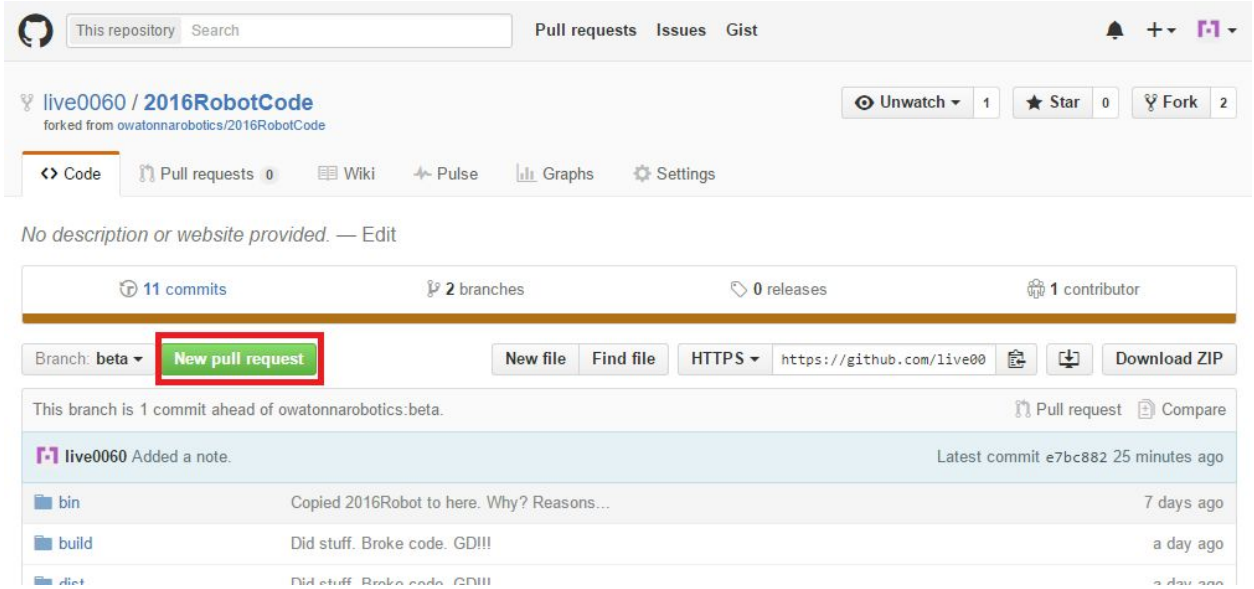


Your changes will now appear in your fork repository on Github! Pretty neat!

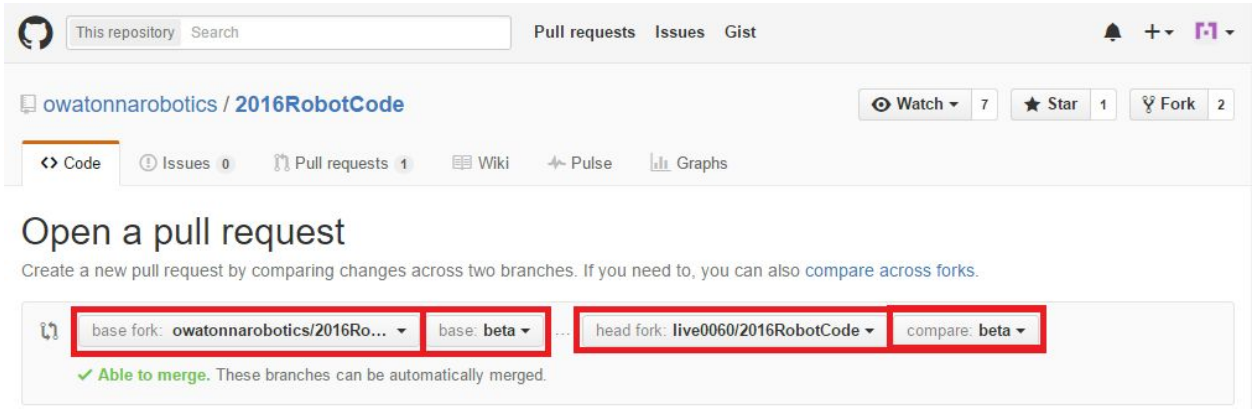


Note: You can make changes, commit, and push to the remote repository as many times as you like. In fact, you should do this often if you are making lots of changes before merging. This way if your hard drive happens to fail, at least some of your changes will be saved on Github's servers.

Next, we need to create a new pull request. Navigate to your fork repository on Github.

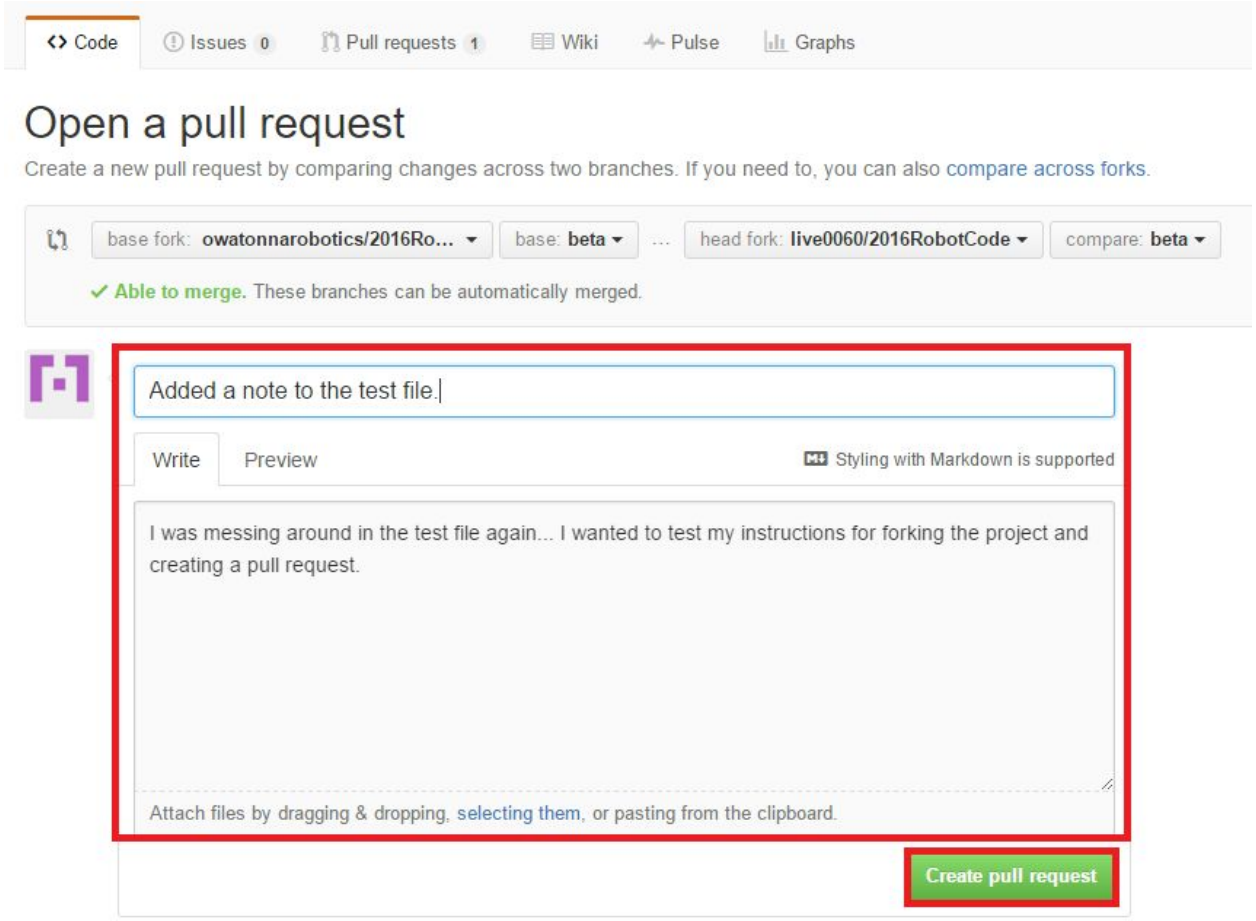


On the “beta” branch, click “New Pull Request”.



Make sure your pull request is merging the beta branch of your fork repository (head fork) with the beta branch of the original project repository (base fork).

You should comment on your merge and can also leave a note for the reviewer.



Click “Create pull request”. A summary of your changes will be sent to the project owner for review. Once they’ve approved your changes, you’ll see them in the original project repository.

Profit!!!