

Rafael Silva – rafael.silva@fareoffice.com

Date: 2017-11-28

Version: 1.0

Document: 024 - Git useful operations - Pt II - 171128

Git useful flows - Pt II

1. Flow 1: Create a new project, upload it to GitHub, add branches to the project, work on your branch, create pull requests on GitHub to merge your branch with master and synchronize the project locally (without merge conflicts).

Steps:

- Create a new project.
- Upload it to github as a first commit.
- Create a new branch on github.
- Fetch the new branch into your local repository.
- Switch to your new branch.
- Set your new local branch to be tracked remotely.
- Work on the project by creating/editing files.
- Push your new local branch to your remote branch.
- Merge your new local branch with your master branch on github through a pull request.
- Pull the changes from your remote repository into your local repository.

Commands and actions in sequence:

- Create a project and a remote repository with the same name.
- Initialize git by opening the terminal, navigating to your right home folder and using the command **git init**.

* work a bit on the project by creating new files, or editing existing ones.
Perform your first commit:

- **git add .**
- **git commit -m "Your commit message!"**

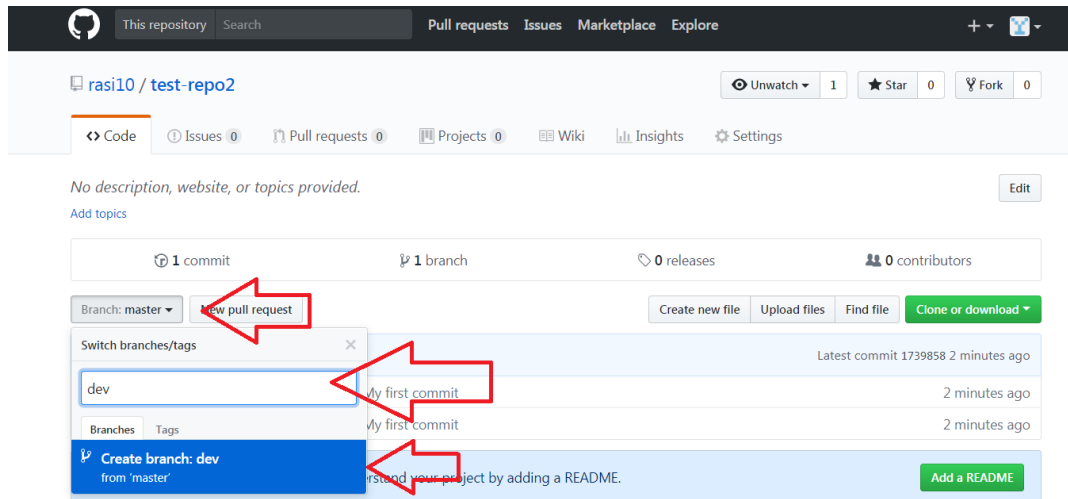
Set the origin to the file with the command:

- **git remote add origin #URL to your repository#**

upload your project with git push:

- **git push -u origin master**

Create a new branch called, e.g. “dev” on github



After that, go back to git bash and use the command:

git fetch --all

After that, check if the branch was fetched. use the command:

- **git branch -a**

After that, switch to your new branch. Use the command:

- **git checkout dev** (Considering that we created a branch called “dev”)

After that, set the branch to be tracked. Use the command:

- **git branch --set-upstream-to remotes/origin/dev**

Check if your branch was set right. Use the command:

- **cat .git/config** (You might be able to see your branch set there)

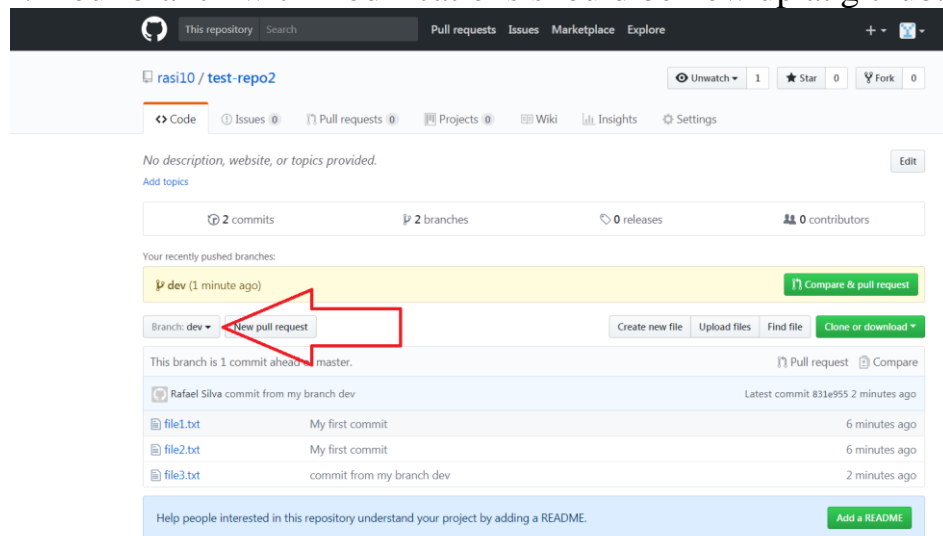
* work a bit on the project by creating new files, or editing existing ones.
Perform a commit from your dev branch.

- **git add .**
- **git commit -m “Commit message”**

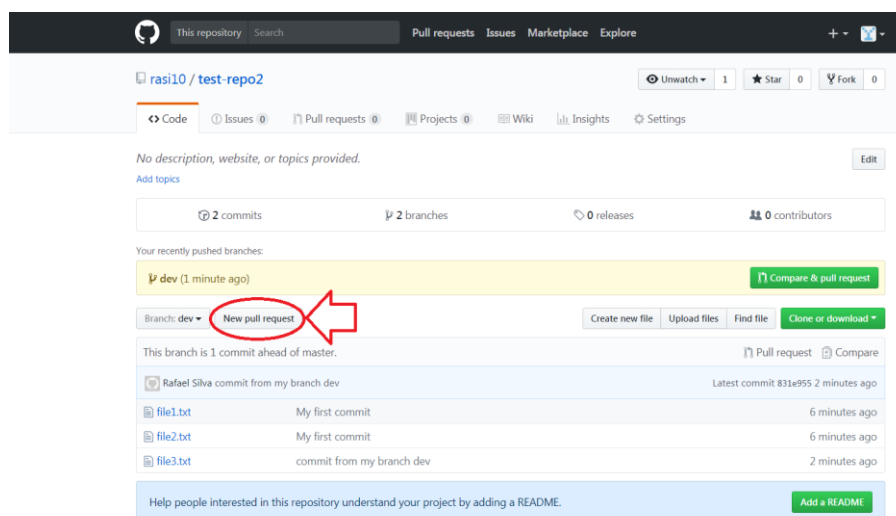
push the dev branch to your remote repository. Use the command:

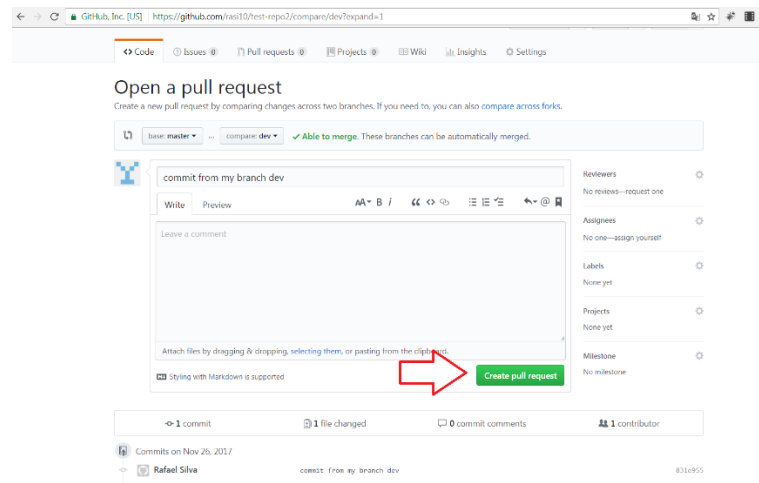
- **git push -u origin dev** (to send the changes from your dev branch)

NOTE: Your branch with modifications should be now up at github.

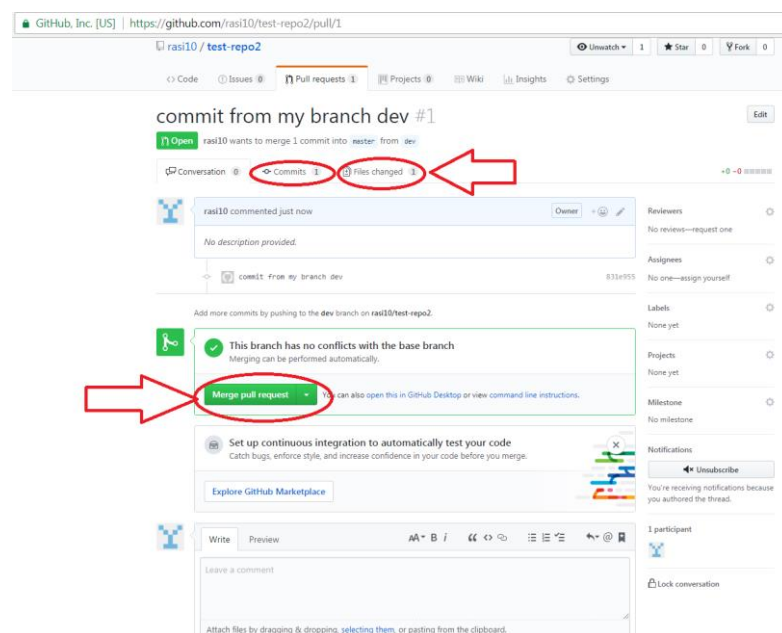


The next step is to make a pull request.





After that, check which changes were made and if they are all ok, click on Merge pull request.



Note that now, your local repository is not synchronized with the remote repository and, thus, you need to make a git pull on the terminal.

Use the command: **git pull origin master**

After that, use **git log** to see that the new merge is included in the logs.

```
rafael@ubt: ~/Desktop/test-repo2
rafael@ubt:~/Desktop/test-repo2$ git pull origin master
remote: Counting objects: 1, done.
remote: Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (1/1), done.
From https://github.com/rasil0/test-repo2
* branch            master       -> FETCH_HEAD
1739858..78f634b  master       -> origin/master
Updating 831e955..78f634b
Fast-forward
 commit 78f634b0c73d74800b29d8330eb83b5a81eeeb80
Merge: 1739858 831e955
Author: rasil0 <mind_traveller@hotmail.com>
Date:   Sun Nov 26 23:51:12 2017 +0100

    Merge pull request #1 from rasil0/dev

    commit from my branch dev

 commit 831e9551aeel7add38e325501b379c14898d26d2
Author: Rafael Silva <rasilva.1986@gmail.com>
Date:   Sun Nov 26 23:39:54 2017 +0100

    commit from my branch dev

 commit 1739858139b51dac11f6ddbdf671625f97be3b0a
Author: Rafael Silva <rasilva.1986@gmail.com>
Date:   Sun Nov 26 23:35:41 2017 +0100

    My first commit
rafael@ubt:~/Desktop/test-repo2$
```

After that, you can see the flow of your project by either using the command:

git log --graph --abbrev-commit --decorate --format=format:'%C(bold blue)%h%C(reset) - %C(bold green)(%ar)%C(reset) %C(white)%s%C(reset) %C(dim white)- %an%C(reset)%C(bold yellow)%d%C(reset)' --all

```
rafael@ubt: ~/Desktop/test-repo2
rafael@ubt:~/Desktop/test-repo2$ git log --graph --abbrev-commit --decorate --format=format:'%C(bold blue)%h%C(reset) - %C(bold green)(%ar)%C(reset) %C(white)%s%C(reset) %C(dim white)- %an%C(reset)%C(bold yellow)%d%C(reset)' --all
* 78f634b - (4 minutes ago) Merge pull request #1 from rasil0/dev - rasil0 (HEAD, origin/master, dev)
* 831e955 - (16 minutes ago) commit from my branch dev - Rafael Silva (origin/dev)
* 1739858 - (20 minutes ago) My first commit - Rafael Silva (master)
rafael@ubt:~/Desktop/test-repo2$
```

Or on github under insights->network

The screenshot shows the GitHub interface for the repository 'rasi10 / test-repo2'. The top navigation bar includes links for 'This repository', 'Search', 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. Below the repository name, there are buttons for 'Unwatch', 'Star', and 'Fork'. The left sidebar contains a list of repository features: Pulse, Contributors, Community, Traffic, Commits, Code frequency, Dependency graph, Network, and Forks. The main content area displays a commit history table with columns for 'Owners' and 'Nov'. A commit by 'rasi10' is visible, dated '26', with a commit hash and a commit message. Red arrows point to the 'Settings' link in the top navigation bar and the 'Network' link in the left sidebar.

Repository: rasi10 / test-repo2

Buttons: Unwatch, Star, Fork

Navigation: Code, Issues, Pull requests, Projects, Wiki, Insights, Settings

Left Sidebar:

- Pulse
- Contributors
- Community
- Traffic
- Commits
- Code frequency
- Dependency graph
- Network
- Forks

Main Content Table:

Owners	Nov
rasi10	26