

GitHub

This repository Search

[Explore](#) [Features](#) [Enterprise](#) [Pricing](#)[Sign up](#)[Sign in](#) [Kunena](#) / [Kunena-Forum](#)[Watch](#)

32

[★ Star](#)

354

[Fork](#)

219

[Code](#)[Issues](#) 97[Pull requests](#) 11[Wiki](#)[Pulse](#)[Graphs](#)

Create a new branch with git and manage branches

Fabian St. George edited this page on Jan 26 · 6 revisions

In your github fork, you need to keep your master branch clean, by clean I mean without any changes, like that you can create at any time a branch from your master. Each time, that you want to commit a bug or a feature, you need to create a branch for it, which will be a copy of your master branch.

When you do a pull request on a branch, you can continue to work on another branch and make another pull request on this other branch.

Before creating a new branch pull the changes from upstream, your master needs to be up to date.

Create the branch on your local machine and switch in this branch :

```
$ git checkout -b [name_of_your_new_branch]
```

Push the branch on github :

```
$ git push origin [name_of_your_new_branch]
```

When you want to commit something in your branch, be sure to be in your branch.

You can see all branches created by using :

```
$ git branch
```

Which will show :

```
* approval_messages
master
master_clean
```

Add a new remote for your branch :

```
$ git remote add [name_of_your_remote]
```

Push changes from your commit into your branch :

```
$ git push origin [name_of_your_remote]
```

Update your branch when the original branch from official repository has been updated :

```
$ git fetch [name_of_your_remote]
```

[▼ Pages](#) 11[Home](#)[Create a new branch with git and manage branches](#)[Developing in PhpStorm with Git](#)[Documentation](#)[Git Issues Guidelines](#)[github](#)[How do i contribute?](#)[How to handle conflicts with git](#)[Overview of branches](#)[Tips with git bash](#)[Useful git commands](#)**Clone this wiki locally**<https://github.com/Kunena/K>[Clone in Desktop](#)

Then you need to apply to merge changes, if your branch is derivated from develop you need to do :

```
$ git merge [name_of_your_remote]/develop
```

Delete a branch on your local filesystem :

```
$ git branch -d [name_of_your_new_branch]
```

To force the deletion of local branch on your filesystem :

```
$ git branch -D [name_of_your_new_branch]
```

Delete the branch on github :

```
$ git push origin :[name_of_your_new_branch]
```

The only difference is the : to say delete, you can do it too by using github interface to remove branch : <https://help.github.com/articles/deleting-unused-branches>.

If you want to change default branch, it's so easy with github, in your fork go into Admin and in the drop-down list default branch choose what you want.

Copyright © 2008 - 2016 Kunena, License: GNU GPL

