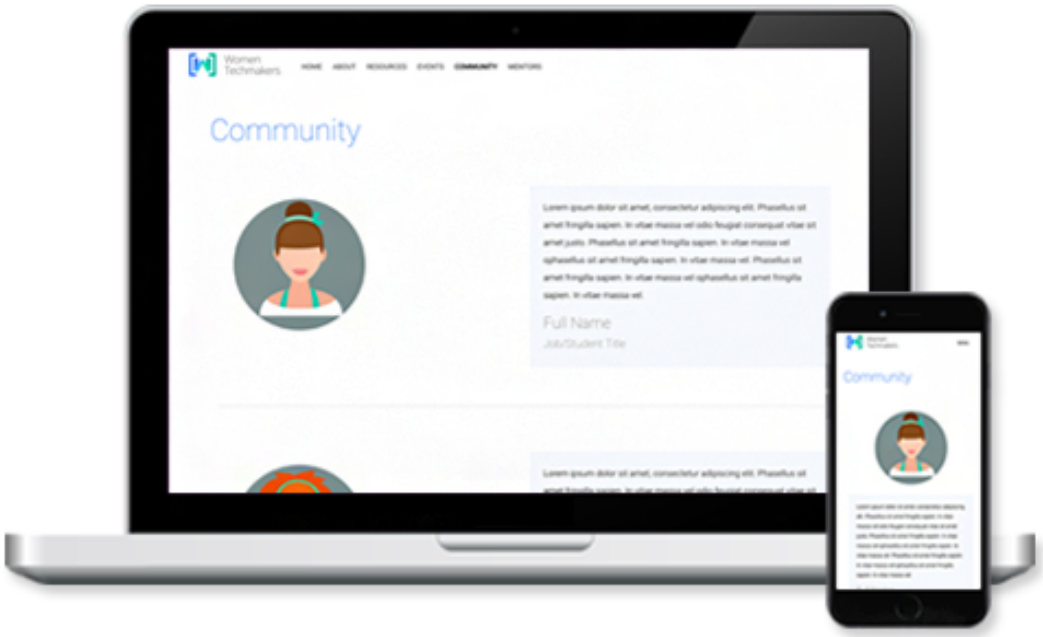
Women Techmakers

Git, GitHub & Web Hosting Workshop
WTM Hamburg

Community Page

Let’s add the content of our community.html page for our WTM website. The end result should look like the following screens:



Get Ready

Let’s make sure we have all we need to add the content of our page.

Install Git (using GitHub for desktop)

1. We would recommend beginners to install the free *GitHub desktop tool*:
   https://desktop.github.com/

   This tool is available for Windows and Mac and includes a command line client. If you want to get a bit more experimental and just work with the command line, follow the instructions here: https://git-scm.com/book/en/v2/Getting-Started-Installing-Git

   We will use the GitHub for desktop application in the following steps. The tool includes Git, our version control system.

2. To use this tool, you need to create a GitHub account:
   https://github.com/join

3. When you've done that, open the GitHub desktop tool and follow these steps to setup your authentication to GitHub:
   https://help.github.com/desktop/guides/getting-started/authenticating-to-github/

4. Once you're finished, you need to follow these other steps to configure Git:
   https://help.github.com/desktop/guides/getting-started/configuring-git-for-github-desktop/

Fork and clone your first Git repository using GitHub for desktop

You will be working on the community page of our WTM HH website by adding the interview questions you collected a few meetups ago.

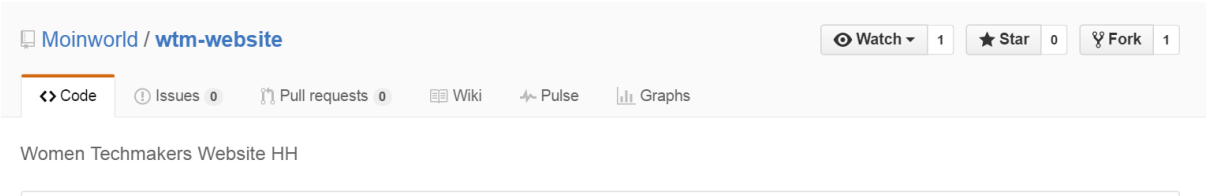
To do that you first need to get an up-to-date version of the website's code.

You can find the code online at:
https://github.com/Moinworld/wtm-website/tree/feature-community

We decided to host our code at **GitHub** to make it accessible to everyone who wants to collaborate on the project.

Follow these steps to create a local copy (a **clone**) of the project:

1) To get your own version of the code you need to log in to the GitHub website, navigate to the WTM repository and press the **Fork** button in the top right corner (see screenshot): https://github.com/Moinworld/wtm-website/tree/feature-community



Forking creates a personal copy of the project and associates it with your account.

2) After the repository has been forked you have to save it locally on your computer to be able to edit the code. To do that, press this button:

You will find it on the left of the “Download ZIP” button.

After clicking the button your browser will prompt a dialogue asking you whether you want to open the GitHub desktop application. Click “App starten” to open the GitHub desktop application.

3) Choose the directory you want to save the repository files to. GitHub for desktop will then **clone** the files from your forked repository onto your hard drive and create a local repository.

Git bash command to clone into the current directory:

```
git clone <url-to-git-repo>
```

4) To check whether everything worked out click the gear icon in the top right corner of GitHub for desktop and click on “Open in explorer”. You should see all files and folders that make up the WTM website project inside your explorer.

Branches

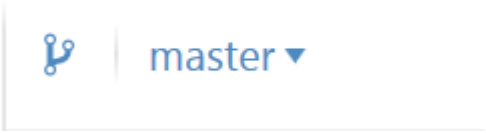
Branching is a very important part of version control with Git.
The standard branch that is created in every Git project is called **master**.

It is not good practice to work directly on the master branch, but to create branches of it and add the changes there. The branch you will be working on is called **feature-community**.

We already created this branch for you. To switch to it inside GitHub desktop click on the arrow icon next to “master” at the top. Select the feature-community branch from the dropdown.

Git bash command to checkout a branch:

```
git checkout <branch-name>
```



The feature-community branch is now checked out and you will make your changes here.

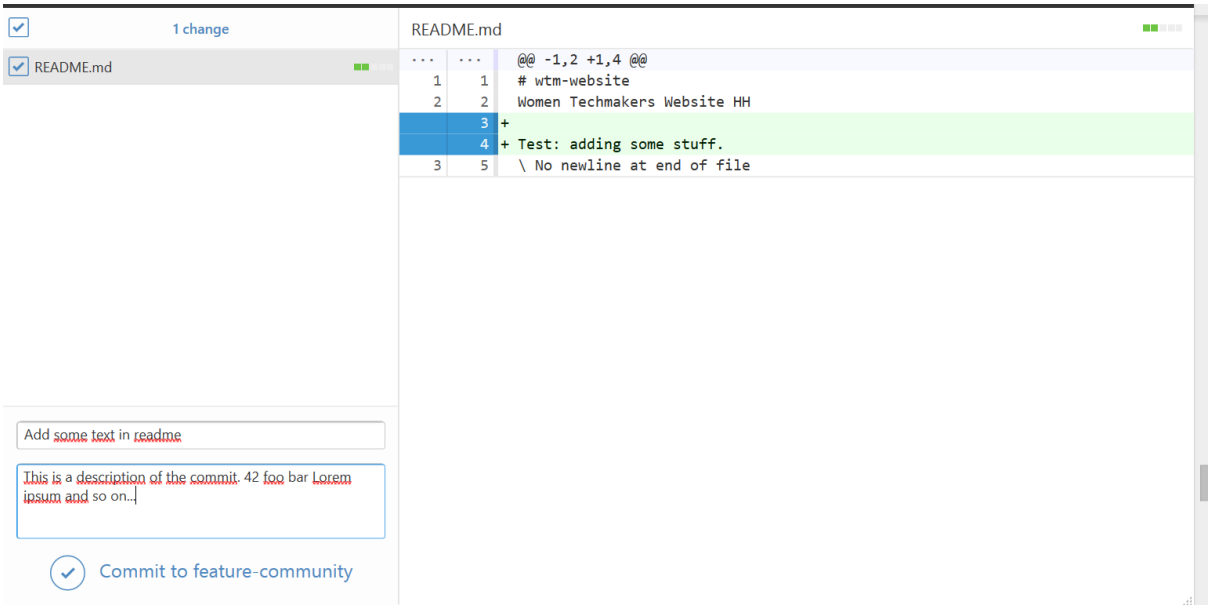
To find out more about branching in git, consult our git-github-webhosting.doc or this url:
https://www.atlassian.com/git/tutorials/using-branches/

Committing changes

If you go ahead and change the content of a file in the repository it will show up in the GitHub desktop application. By clicking on the filename on the left you can see what has changed inside the file.

You can directly commit the changes by typing in a commit summary and description. To commit the changes press the button at the bottom left.

The screenshot shows edits inside the README.md file and the commit message.



If you are working with Git Bash you have to add your changes before you can commit them. You can use the status command to show the status of your files first:

```
// show status of files in repository
git status

// Output by git will be something like:

On branch feature-community
Your branch is up-to-date with 'origin/feature-community'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)

        <filename>

nothing added to commit but untracked files present (use "git
add" to track)

// add the files you want to commit
git add <filename>

// commit with a message
git commit -m “Add empty text file”
```

Pull and push the changes

To make sure you're up to date with the newest version of the code, you should use the Sync button in the GitHub desktop tool (top right).

This will initiate a pull, which integrates the changes from the remote repository into your local one. Afterwards, it pushes your local changes to the remote.

Git bash command to pull and push changes:

```
// pulling then pushing
// <remote> is called 'origin' by default
git pull <remote> <branch>

git push <remote> <branch>
```

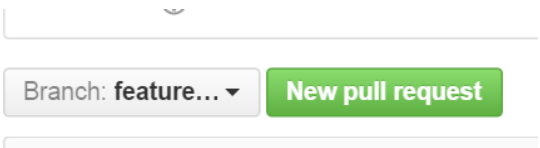
If you want to do a complete sync like the one in the GitHub tool, you have to follow these instructions. Beware that it's quite advanced and involves merging branches:
https://help.github.com/articles/syncing-a-fork/

Creating a pull request

After you have made and committed your changes on the feature-community branch you can create a **pull request** that notifies the owners of the original repository (your meetup organisers) of your changes. They will then be able to incorporate your changes into the existing code**.**

You can do this either through the GitHub desktop application by clicking on the Pull Request button in the top right corner.

You can also log in to GitHub online and do it through their user interface.



Happy Coding!

Women Techmakers HH Team