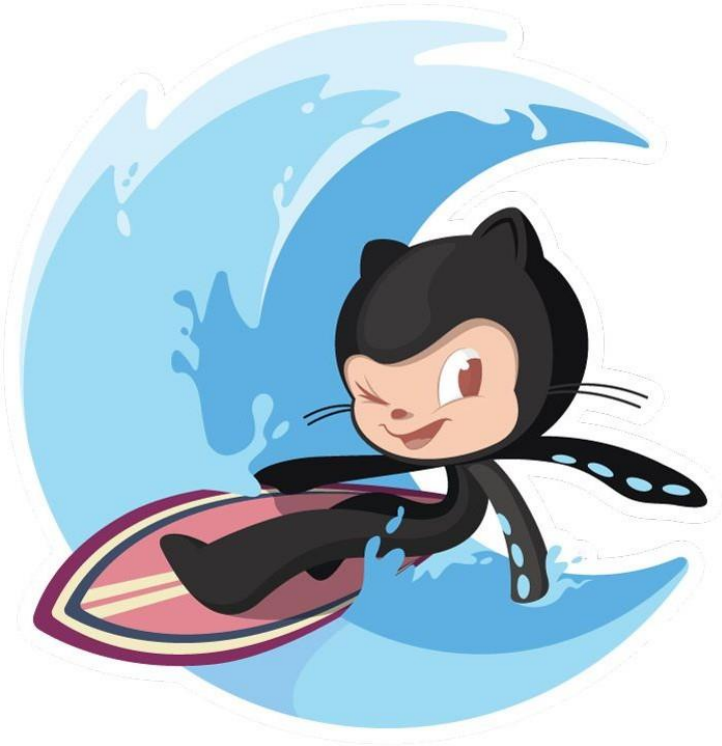
Mark McFadden
https://m2web.github.io/
https://github.com/m2web

Introduction to GitHub

Mark McFadden
https://m2web.github.io/
https://github.com/m2web

Presentation Flow

- What is GitHub?
- GitHub Fundamentals 101



So, What is GitHub?

GitHub is the single largest host for Git repositories and is the central point of collaboration for millions of developers and projects.

History:

Know exactly which files changed, who made those changes, and when those changes occured.

Backup:

Ability to have different versions of the code in different places.

Collaboration:

Collaborate easily with other people on the same project by uploading and receiving changes

GitHub is Good!

The fundamentals of GitHub 101.

Review from Git Intro

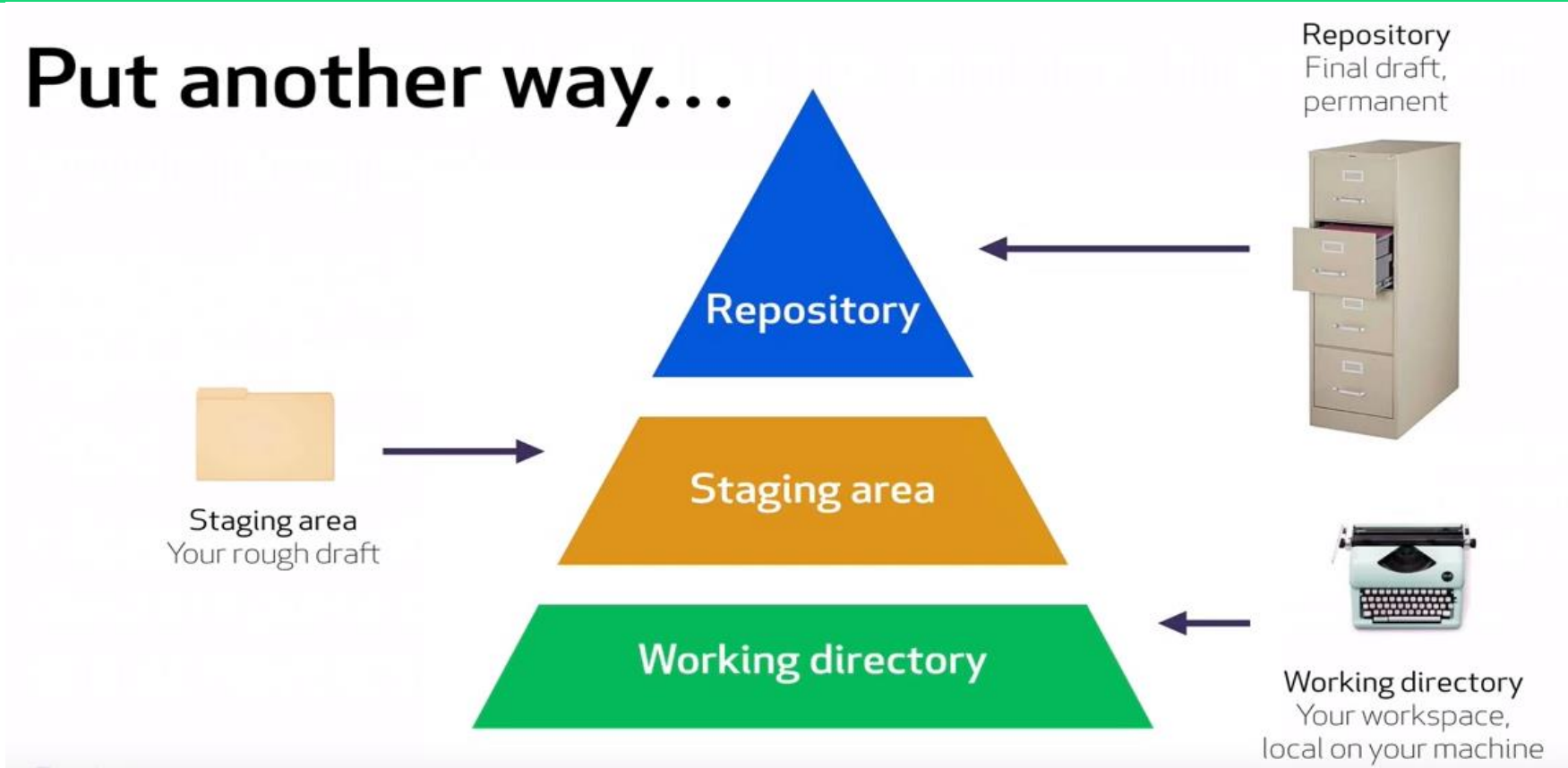
Git is like a desk

Working directory
where you write

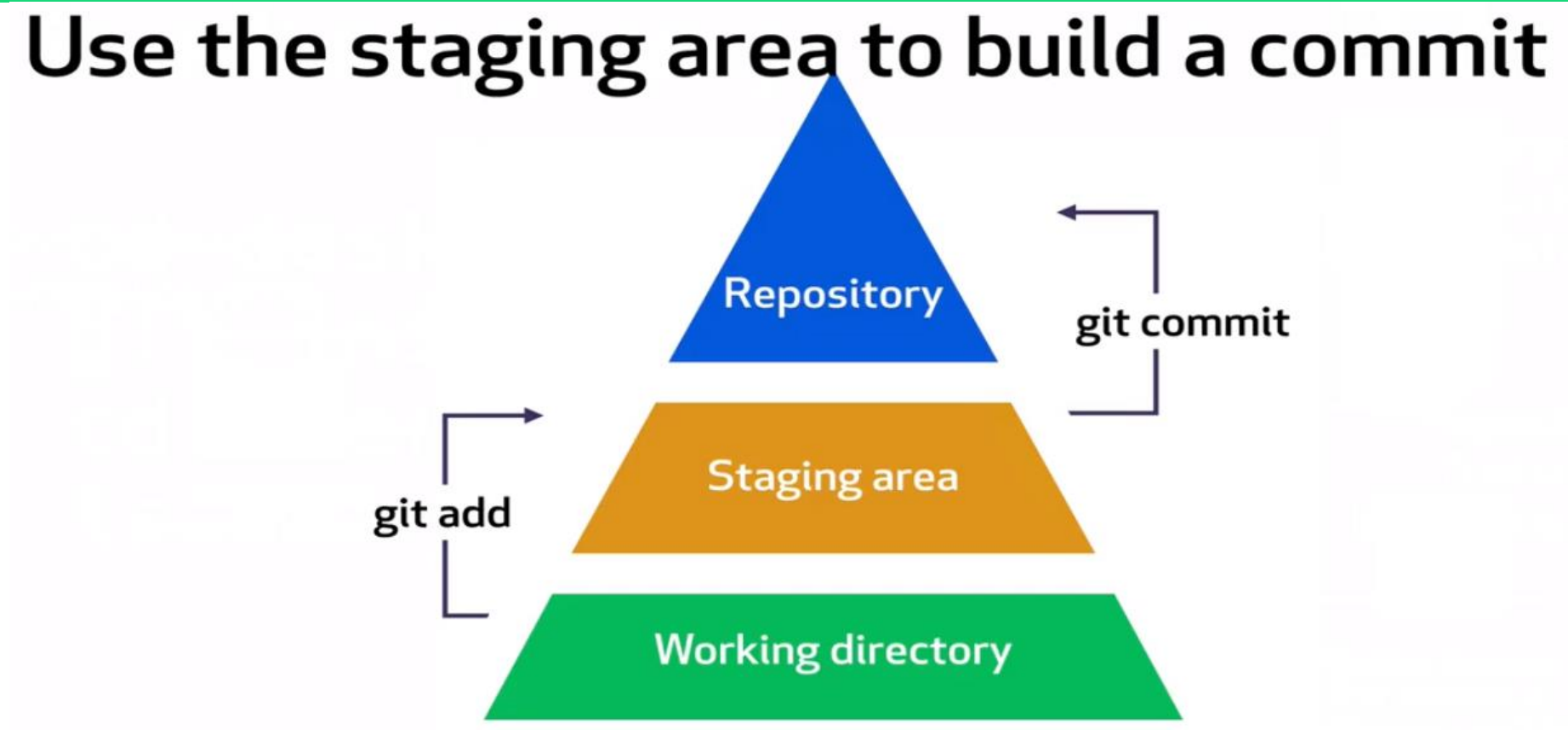
Staging area
rough draft, in a
manila folder

Repository
final draft
in the filing cabinet

Review from Git Intro



Review from Git Intro



Initializing a new repository

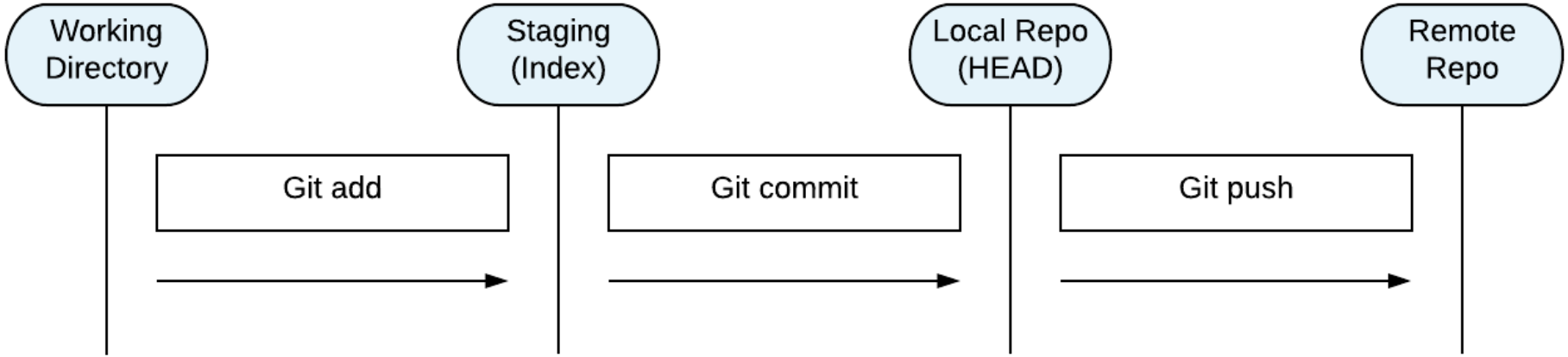
```
# creating a new folder for our project
$ mkdir MyProject
# changing directory to our project folder
$ cd MyProject
# initializing the current folder as a repository
$ git init

Initialized empty Git repository in /home/user/MyProject/.git/
```

Commits

Checkpoints/Snapshot of the state of your repository (project) at a particular time.

Git Flow Elements

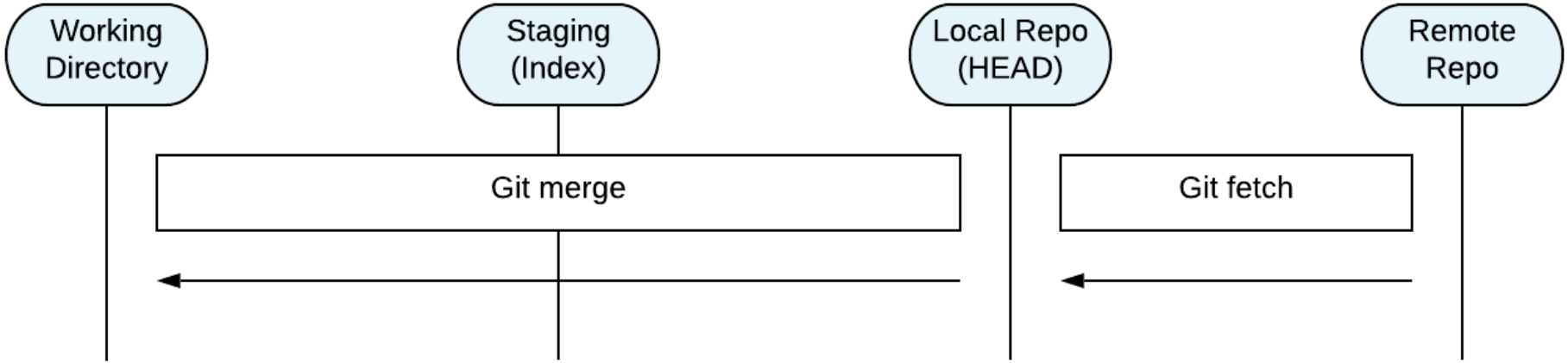


`git add` is a command used to add a file that is in the working directory to the staging area.

`git commit` is a command used to add all files that are staged to the local repository.

`git push` is a command used to add all committed files in the local repository to the remote repository. So in the remote repository, all files and changes will be visible to anyone with access to the remote repository.

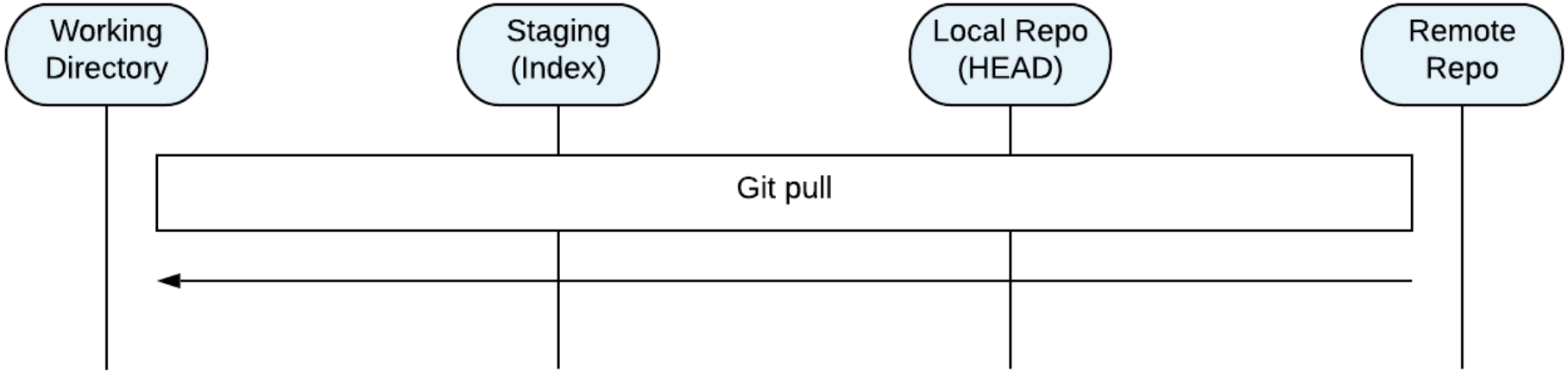
Git Flow Elements



`git fetch` is a command used to get files from the remote repository to the local repository but not into the working directory.

`git merge` is a command used to get the files from the local repository into the working directory.

Git Flow Elements

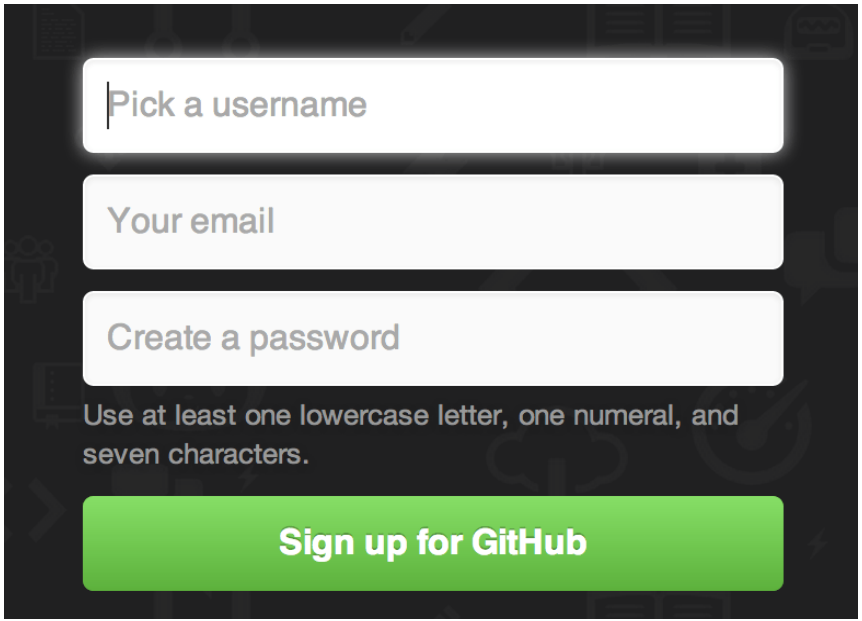


`git pull` is command used to get files from the remote repository directly into the working directory.

It is equivalent to a `git fetch` and a `git merge`.

GitHub Account Setup

The first thing you need to do is set up a free user account. Simply visit https://github.com, choose a username that isn't already taken, provide an email address and a password, and click the big green "Sign up for GitHub" button.



Contributing to a Project – As a Collaborator

The GitHub Flow

GitHub is designed around a particular collaboration workflow, centered on Pull Requests created from Topic Branches. Here's how it generally works:

1. As a collaborator Clone the project. If you do not have push access to the project you can Fork the project.
2. Create a topic branch from master.
3. Make some commits to improve the project.
4. Push this branch to your GitHub project.
5. Open a Pull Request on GitHub.
6. Discuss, and optionally continue committing.
7. The project owner merges or closes the Pull Request.
8. Sync the updated master back to your repo.

see https://git-scm.com/book/en/v2/GitHub-Contributing-to-a-Project

Contributing to a Project – Fork a Project

Forking Projects
If you want to contribute to an existing project to which you don't have push access, you can "fork" the project.

To fork a project, visit the project page and click the "Fork" button at the top-right of the page.

After a few seconds, you'll be taken to your new project page, with your own writeable copy of the code.

see https://git-scm.com/book/en/v2/GitHub-Contributing-to-a-Project

Contributing to a Project – Not a Collaborator

The GitHub Flow

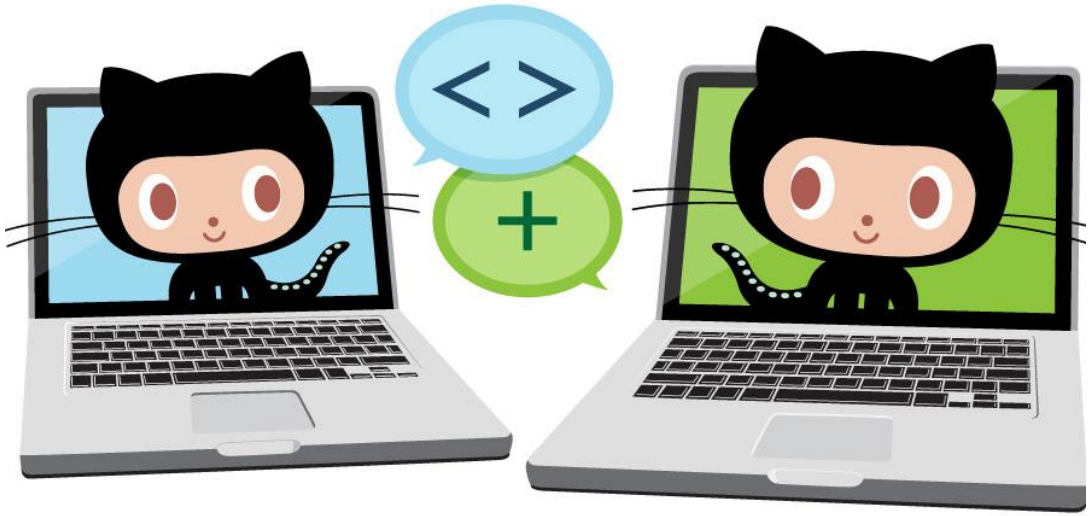
GitHub is designed around a particular collaboration workflow, centered on Pull Requests created from Topic Branches. Here's how it generally works:

1. Fork the project.
2. Create a topic branch from master.
3. Make some commits to improve the project.
4. Push this branch to your GitHub project.
5. Open a Pull Request on GitHub.
6. Discuss, and optionally continue committing.
7. The project owner merges or closes the Pull Request.
8. Sync the updated master back to your repo.

see https://git-scm.com/book/en/v2/GitHub-Contributing-to-a-Project

DEMO: Contributing to a Repo

Questions ?



Mark McFadden
mcfaddenm1@nku.edu
https://m2web.github.io/
https://github.com/m2web