Version Control - Github

Learning objectives

- Working in teams
- Know what Version Control is and why it is used
- Knowledge of Git
- Github account creation
- Create your first repository
- Update your repository remotely

Teamwork?

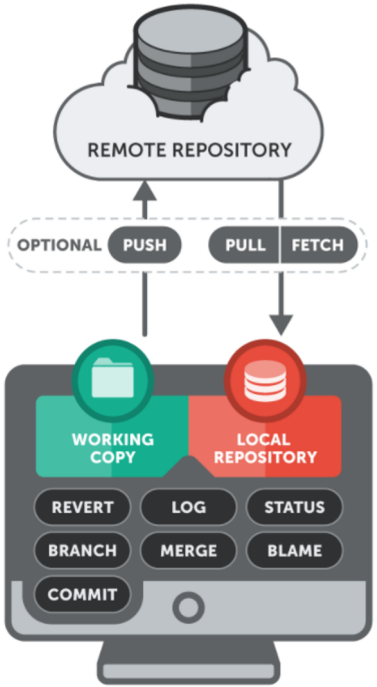
Version control - Git

Version control systems are a category of software tools that help a software team manage changes to source code over time. Version control software keeps track of every modification to the code in a special kind of database. If a mistake is made, developers can turn back the clock and compare earlier versions of the code to help fix the mistake while minimizing disruption to all team members.

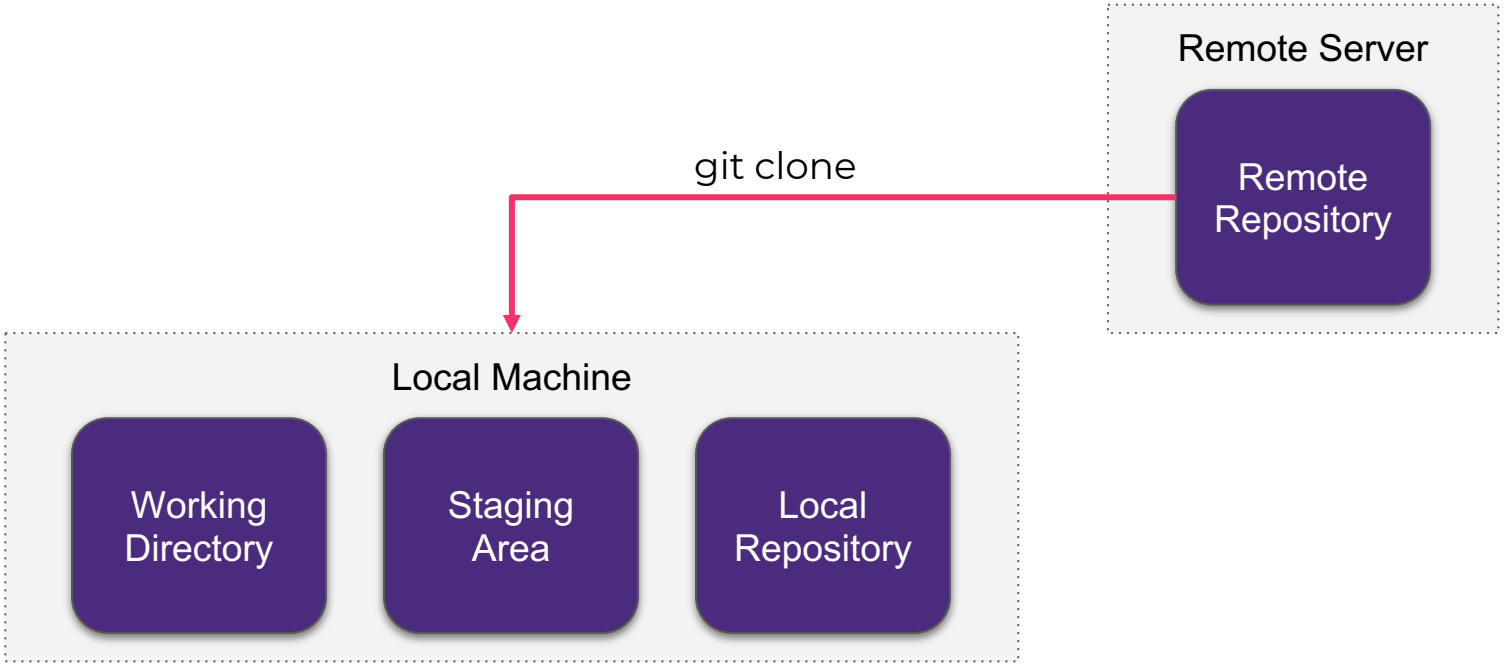
- Easier to share code between multiple developers (collaboration and code reviews)
- More robust code management (backup / reversion control)
- Git is an example of a distributed version control system
- We use Github which is a managed service to host our git version control

Overview

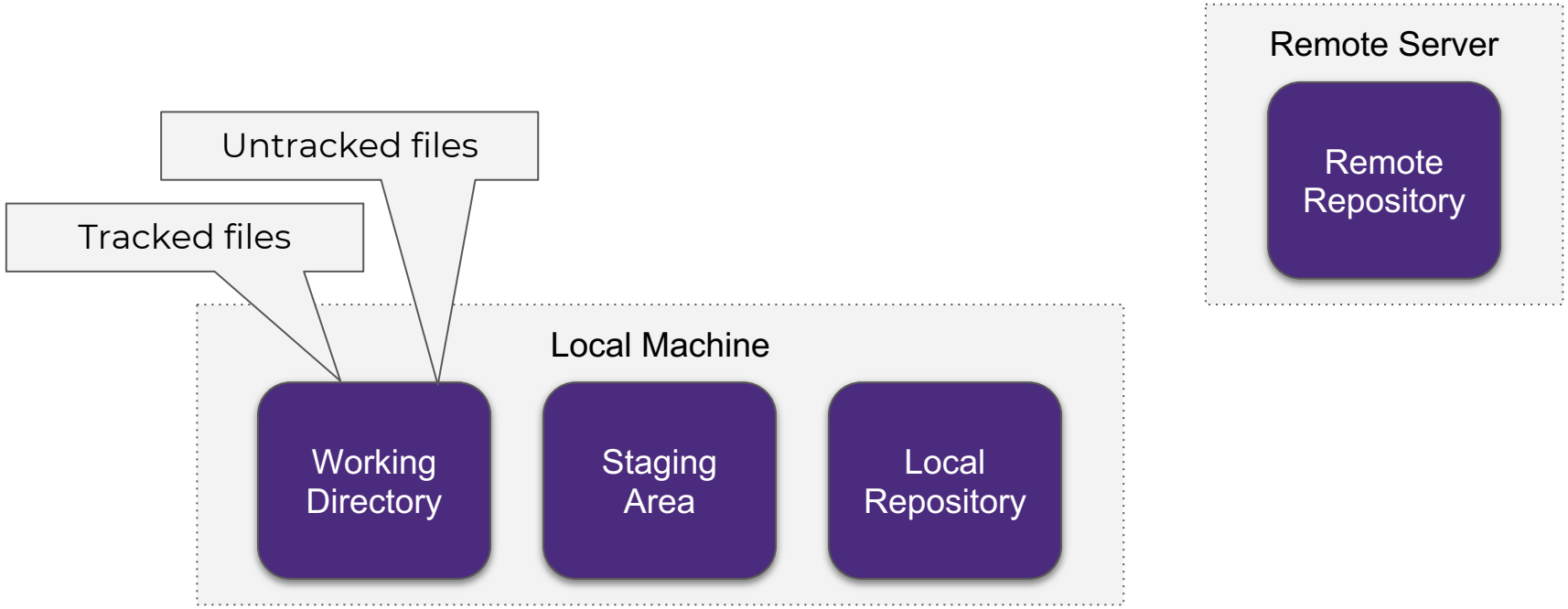
GIT



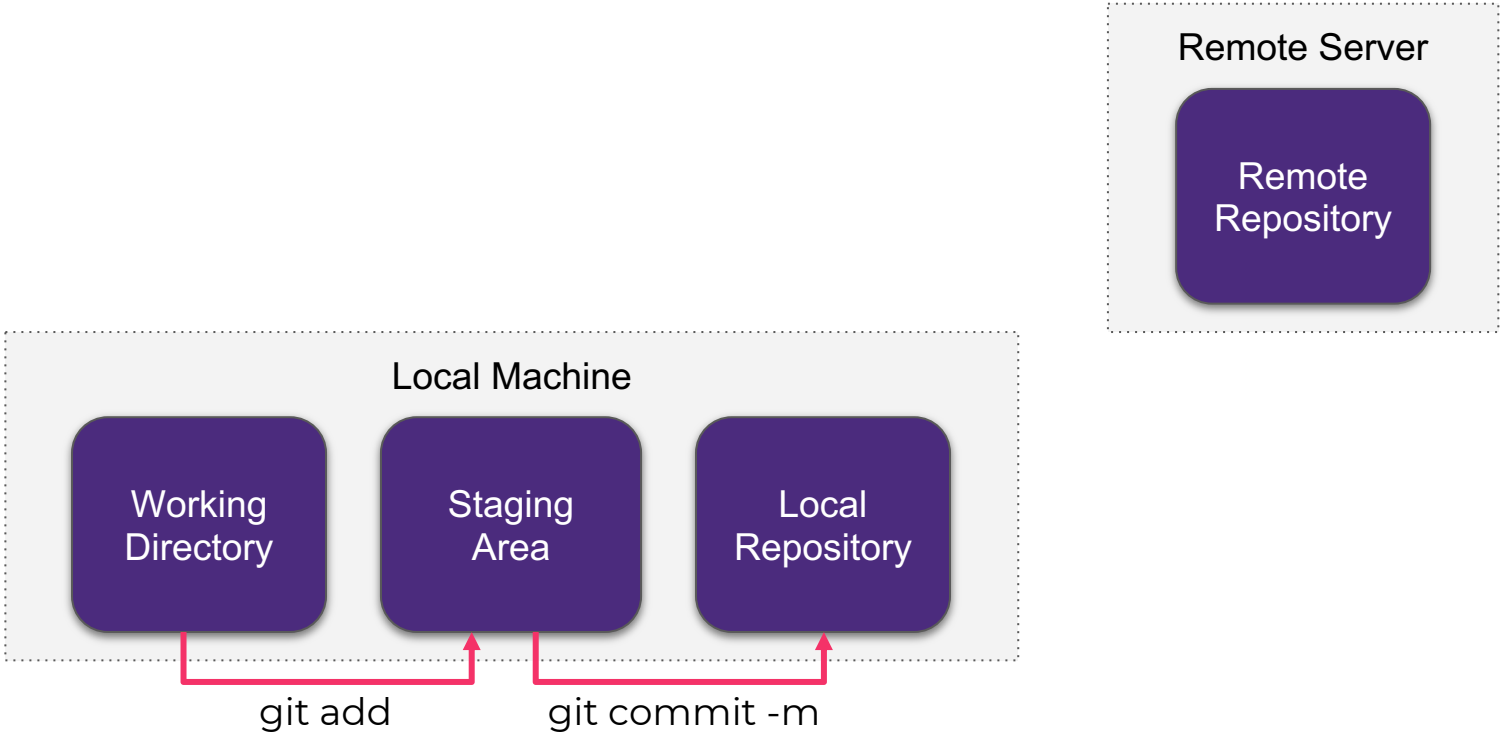
Cloning a Git repository



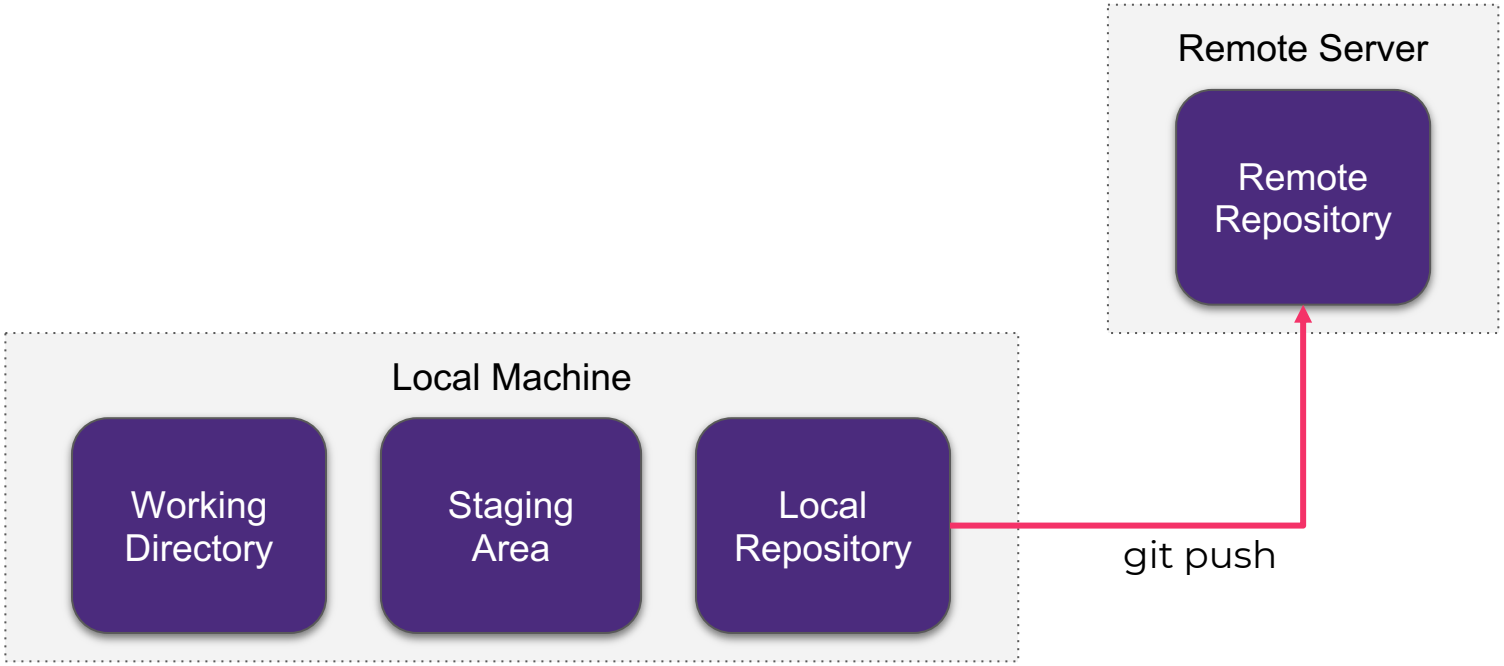
Make changes to the code in your working directory



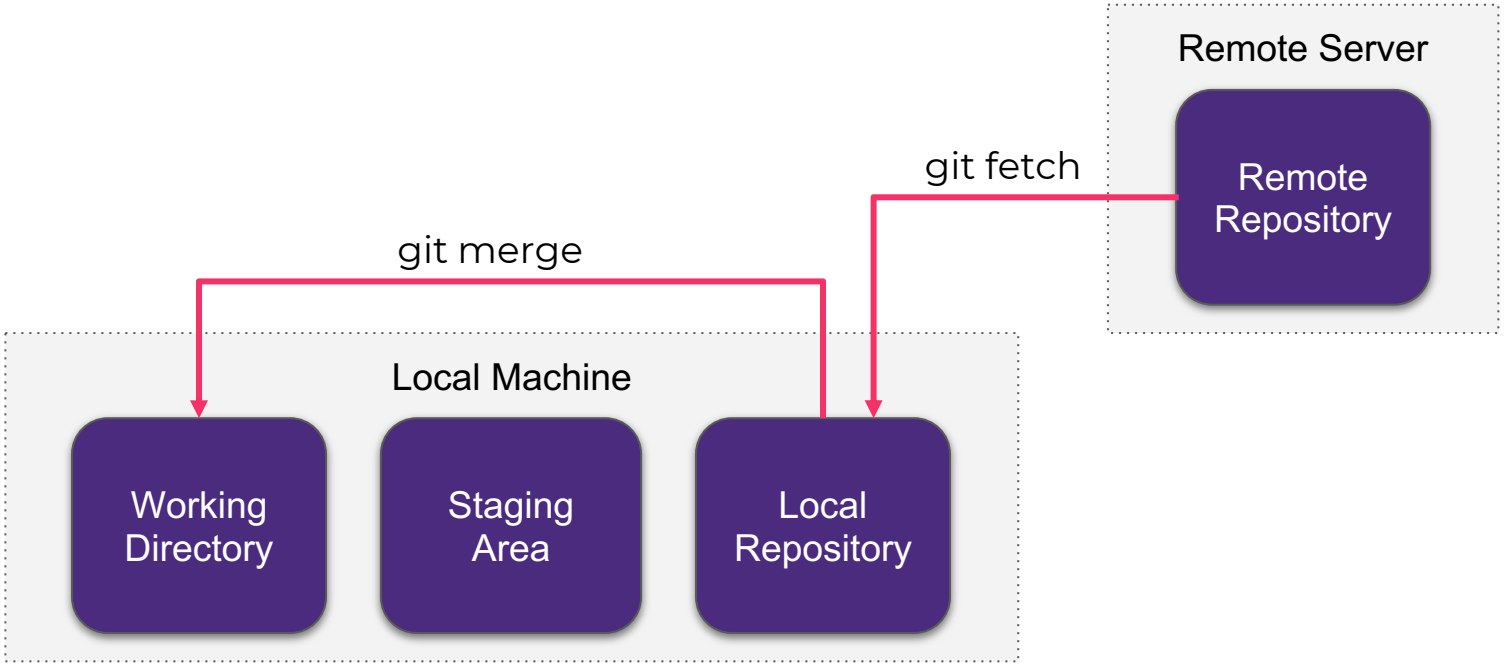
Add and commit your changes



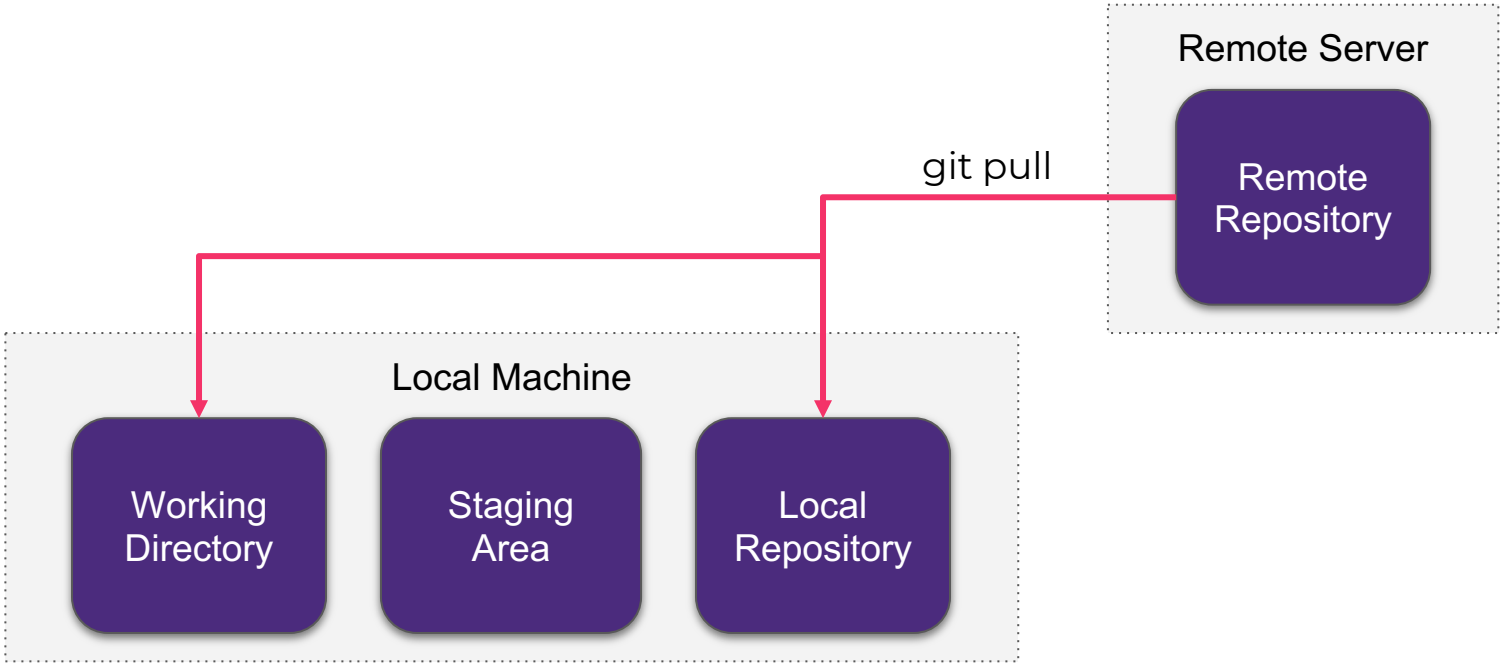
Push your changes



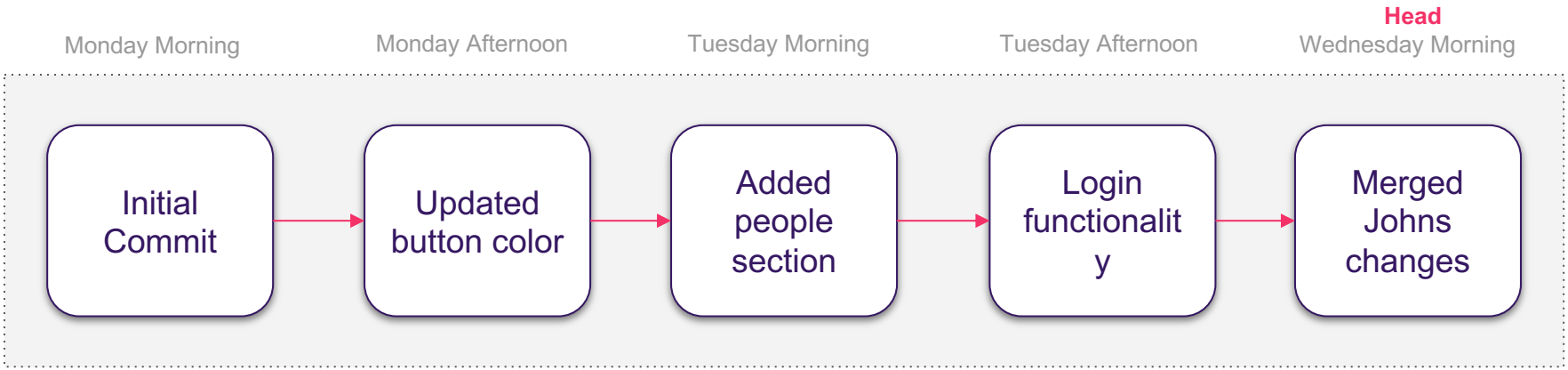
Fetching changes from the remote repository



Pulling changes from the remote repository

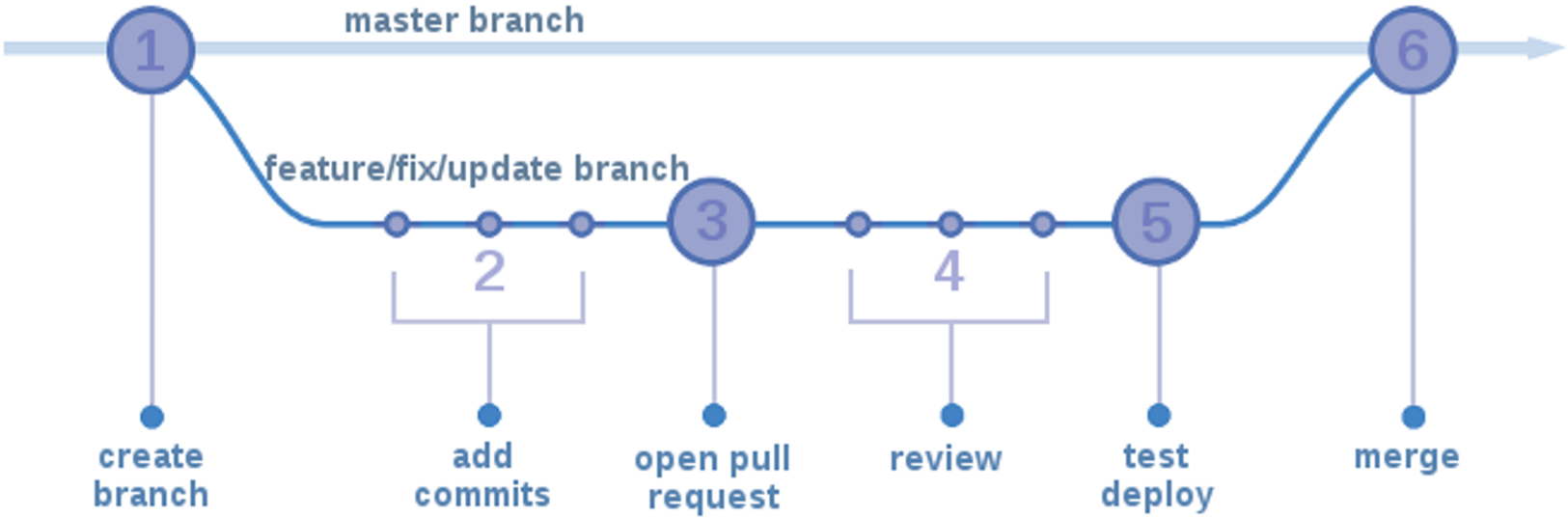


History


Monday Morning
Monday Afternoon
Tuesday Morning
Tuesday Afternoon
Head
Wednesday Morning
Initial Commit
Updated button color
Added people section
Login functionality
Merged Johns changes
git add *
git commit -m “initial commit”
git add buttons.css
git commit -m “updated button color”
git add index.html
git commit -m “added people section”
git add login.html & git add auth.js
git commit -m “login functionality”
git pull
… Manually merge …
git commit -m “merged johns changes”

Challenge

1. Create a new repository on your machine called ‘first-git-repo’
2. Create an index.html file and add your name to it.
3. Add the new untracked file (*git add index.html*)
4. Commit the changes (*git commit -m ‘my first commit’*)
5. Create a second file of any kind and repeat the git add/commit process.


master branch
1
6
feature/fix/update branch
3
5
2
4
create branch
add commits
open pull request
review
test deploy
merge

Challenge 2

1. Create a repository on Github using your new username called “nology-coursework”
2. Clone the repository down inside your “nology” directory
3. Create a directory called “01-Tooling” inside the repository
4. Inside that, create two directories, one for “01-CommandLine” and one for “02-VersionControl”
5. Add your kennels directory from the Command Line lesson into “01-CommandLine”
6. Inside ”02-VersionControl”, create a README.md file that says “It works!”
7. Push your changes and view them on Github

Additional resources

- https://git-scm.com/
- https://github.com/
- https://guides.github.com/introduction/flow/
- https://www.git-tower.com/learn/git/ebook/en/desktop-gui/appendix/from-subversion-to-git