

CAS 741/CES 741

Dr. Spencer Smith

McMaster University

Git and GitLab/GitHub

Revised: September 7, 2017

This tutorial will introduce GitLab, GitHub, Git and issue tracking.

Components of Lab

1. Introduction to GitLab/GitHub
2. GitLab Exercises

Details

- Make sure that you have Git on your machine. If you do not have Git, download it from:
<https://git-scm.com/>
- Try the interactive git tutorial available at: <https://try.github.io>
- Work through the Software Carpentry lab exercise on Git. This is available at <http://swcarpentry.github.io/git-novice/>
- Great resource for git help:
<https://git-scm.com/book/en/v2/Git-Basics-Getting-a-Git-Repository>
- Learn about branches and how/when to use them:
<https://git-scm.com/book/en/v2/Git-Branching-Branches-in-a-Nutshell>
- View some videos on Git: <https://git-scm.com/videos>
- Learn about a `.gitignore` file:
<https://help.github.com/articles/ignoring-files/>
Note: Generated files **should not** be placed into version control. The only exception we recommend is for \LaTeX generated pdf files. For convenience, they should often be under version control.
- Read through the tutorial on git workflows:
<https://www.atlassian.com/git/tutorials/comparing-workflows/>

- Once you are comfortable with Git, set-up your project repo on GitHub. We are using GitHub so that it is easy to share your project. Creating a repo will involve the following steps
 1. Create a new public repository on GitHub
 2. Add the members (collaborators), including the instructor
 3. You may later have to add some classmates
 4. You may want to add your supervisor and/or some colleagues
 5. Create the initial folder structure for your project, as given in the BlankProjectTemplate folder.
 6. Create your .gitignore file and ignore .aux files as an example. Commit this file to your repository.
- Review the material on issue tracking also available in this folder, in the file instructions_issue_tracking.pdf.