Introduction to Version Control Systems 2

DATA 4010 Seminar – Fall 2023

Stefano Ansaloni

University of Manitoba

September 25, 2023

University of Manitoba

Digital Research Alliance of Canada

About me

Stefano Ansaloni

Cloud Computing Specialist at University of Manitoba
(part of the HPC support team)

Software Developer and DevOps Specialist since 2017

Linux User/Admin since 2005

University of Manitoba

Refresh – What is a Git remote?

A *remote* (or *remote repository*) is a copy of your local repository hosted on the Internet or network somewhere.

Remote repositories are used to ease the collaboration between multiple developers by pushing and pulling data to and from them when you need to share work.

Code hosting platforms

GitHub	https://github.com
GitLab	https://gitlab.com
Bitbucket	https://bitbucket.org
Gitea	https://gitea.com
Codeberg	https://codeberg.org

Note: remember that is your responsibility to read, understand, and decide whether to accept all the policies/user agreements of the service(s) you use.

University of Manitoba

More than just remote repositories

In addition to remote repositories, those services usually provide:

- access control
- bug tracking
- task management
- continuous integration (CI)
- continuous delivery/deployment (CD)
- wiki
- ... and more

GitHub

GitHub is a common choice to host code for public open-source projects, or to create private repositories for personal/non-public projects.

It is also the world's largest source code host as of June 2023:

- over 100 million developers
- more than 372 million repositories
- $\sim$ 28 million public repositories

GitHub will require all users who contribute code, to enable one or more forms of two-factor authentication (2FA) by the end of 2023.

Wikipedia [GitHub]
GitHub [2FA first announcement]
GitHub [2FA second announcement]

University of Manitoba

Creating a new GitHub repository

1. In the top-right corner of any page, use the "+" drop-down menu, and select "*New repository*"
2. Type a short name for the new repository (e.g. "first-repo")
3. Choose the repository visibility: "*Public*" or "*Private*"
4. Review the creation options
5. Click "*Create repository*"

After creating a new GitHub repository it can be added as a remote to a local git repository.
(e.g. "`git remote add some-name <new_github_repo_url>`")

Adding collaborators to a GitHub repository

1. Navigate to the main page of the desired repository
2. Under the repository name, click "*Settings*"
3. In the "Access" section of the sidebar, click "*Collaborators*"
4. Click "*Add people*"
5. Use the search field to search and select a GitHub user
6. Click "*Add <user_name> to <repo_name>*"
7. The selected user will receive an invite email to the repository (once they accept the invitation, they will have collaborator access to the repository)

Forking a GitHub repository

A *fork* is a copy of an existing repository that shares code and visibility settings with the original upstream repository.

1. Navigate to the existing repository to be forked
2. In the top-right corner of the page, click "*Fork*"
3. Review the forking options
4. Click "*Create fork*"

Creating a GitHub pull request

A *pull request* (or *PR*) is a way to ask an upstream repository to integrate changes pushed to a repository that is a fork of the main one.

Once a *pull request* is opened, maintainers can discuss and review the potential changes with collaborators and add follow-up commits before the changes are merged into the main repository.

GitHub [About pull requests]
GitHub [Creating a pull request]

University of Manitoba

Creating a GitHub issue

GitHub allows users to create and manage *issues* to track ideas, feedback, tasks, or bugs.

1. Navigate to the main page of the desired repository
2. Under the repository name, click "*Issues*"
3. Click "*New issue*"
4. In the "*Title*" field, type a meaningful title for the issue
5. In the comment body field, type a description of the issue
6. Review the other options
7. Click "*Submit new issue*"

GitHub [About issues]
GitHub [Creating an issue]

University of Manitoba

Extra - Creating a GitHub gist

A *gist* is a special type of GitHub repository intended to provide a simple way to share code snippets with others (like a pastebin service).

1. Login to GitHub and navigate to https://gist.github.com
2. In the "*Filename including extension*" field, type a file name for the gist (including the file extensions)
3. In the file contents field, type the text of the gist
4. Click "*Create secret Gist*" or "*Create public gist*"
   (note: secret gists are not private, i.e. anyone who has the gist's url can access it)

Extra - GitHub Pages

GitHub Pages is a static web hosting services offered by GitHub that can be used to serve websites.

It takes HTML, CSS, and JavaScript files straight from a repository and serves them using a "`github.io`" subdomain.

GitHub Pages can be integrated with Jekyll (https://jekyllrb.com) to generate the website from plain text files.

University of Manitoba

GitHub [About GitHub Pages]

What is a license file

A *license* (or *software license*) is a legal document governing the use or redistribution of software.

Usually the *license* is a text file that resides in the root directory of a project (e.g. "`LICENSE`", "`LICENSE.txt`", ...).

When you own the source code of a software, you can choose the license type that best suits your needs.

`Wikipedia [Software license]`
`GitHub [Licensing a repository]`
`Choose-a-license website`

University of Manitoba

What is a readme file

A *readme file* (or *README*) is a text file that resides in the root directory of a project, containing the following information:

- the purpose of the project
- instructions on how to build/install the project
- guidance on how to use the project
- how to contribute to the project

By ensuring that the *readme file* is comprehensive, developers can better collaborate and maintain the project.

Medium [Why having a Readme]
GitHub [About READMEs]

University of Manitoba

Formatting a readme file

Since a readme file is just a text file, it does not allow any special formatting by itself.

However, GitHub can render readme files when they are written using the Markdown syntax, and have the "`.md`" extension (e.g. "`README.md`").

It is also important to note that GitHub offers an extended version of the basic Markdown syntax, and you can use it when writing readmes, comments and issues.

Markdown Guide

GitHub [Basic writing and formatting syntax]

University of Manitoba

Markdown cheat-sheet

Heading	`# Biggest Header`
	`...`
	`###### Smallest Header`

Bold	`**bold text**`
Italic	`*italicized text*`
Blockquote	`> quoted text`
Unordered list	`- item 1` `- item 2`
Ordered list	`1. item 1` `2. item 2`
Link	`[link title](https://github.com)`
Inline code	`` `code` ``
Multiline code	` ```` ` `code` `more code` ` ```` `

Table	`| col 1 hdr | col 2 hdr |`
	`| --- | --- |`
	`| some text | other text |`

University of Manitoba

Markdown Guide [Cheat Sheet]

GitHub Markdown cheat-sheet

Mentioning people or teams	`@username` `@organization/team_name`

Referencing issues or pull requests	`#issue_or_pr_number`
	`username/repo_name#issue_or_pr_number`
	`organization/repo_name#issue_or_pr_number`

Referencing commits	`commit_sha`
	`username@commit_sha`
	`username/repo_name@commit_sha`
	`organization/repo_name@commit_sha`

GitHub [Basic writing and formatting syntax]
GitHub [Autolinked references and URLs]

University of Manitoba

Questions?

University of Manitoba

Thank you

University of Manitoba