



# GitLab

Lara Ianov, Ph.D.

Neurodevelopmental Bioinformatics Initiative (NBI)

Civitan International Research Center

University of Alabama at Birmingham

# Git



- Version control software
  - Tracks changes among all members
  - Maintains a history of all changes in an organized way and allows one to revert changes
  - A higher order method over the standard "copy and change name method": copy a file with changes to it and name it differently or place it in time-stamped directories - usually these scripts are filled with unnecessary comments documenting the changes
  - A git repository contains the full history of the project. Each clone will always mirror the full repository. Thus, git is a Distributed Version Control System

# GitLab, GitHub, Bitbucket etc.



- Web-based services which can be used to host your git repositories in a streamlined manner. Here are some differences among them:

Free Plans	Public Repos	Private Repos	Collaborators	Storage Space	Hosting	Support
GitHub Public	Unlimited	* 0	Unlimited	N / A	Cloud	Email / Forum
Bitbucket Small teams	Unlimited	Unlimited (1Gb /project)	5	N / A	Cloud	Email / Forum
GitLab Cloud Hosted	Unlimited	Unlimited (10Gb / Project)	Unlimited	Unlimited	Cloud	Forum
GitLab Community Edition	Unlimited	Unlimited	Unlimited	N / A	Self-hosted	Forum

\* As a student (with .edu account), you can have private repos for free with access to “student developer plan”. Details at: <https://help.github.com/articles/applying-for-a-student-developer-pack/>



UAB RC hosts



# GitLab


- Sign-in with Blazer ID and password: [https://gitlab.rc.uab.edu/users/sign\\_in](https://gitlab.rc.uab.edu/users/sign_in)
- Learning sources:
  - Git for beginners from UAB RC: [https://docs.uabgrid.uab.edu/wiki/Git\\_For\\_Beginners](https://docs.uabgrid.uab.edu/wiki/Git_For_Beginners)
  - GitLab tutorial: <https://www.tutorialspoint.com/gitlab/index.htm>
  - GitLab docs: <https://docs.gitlab.com/>
  - Atlassian's tutorial: <https://www.atlassian.com/git/tutorials/what-is-version-control>
  - Google 😊
- **Today's focus:** GitLab's graphical interface (with some emphasis on important commands)
- Basic git terminology:
  - `Clone`: copy a repository (repo)
  - `Commit`: record changes to the repo
  - `Push`: updates remote git repo with committed changes (if working outside of GitLab environment)
  - `Branch`: independent area of development from the primary set/branch
  - ... and more



# GitLab groups


**CIRC\_NBI**   
Group ID: 249



 

New project 







This group contains the NBI hosted pipelines and supplementary scripts



**Subgroups and projects**   Shared projects   Archived projects

Most stars 



  **L** **Labs** Owner



Lab specific scripts

 0    1    1         



  **M** **Matt\_Alexander**



Script(s) by the NBI in support of Dr. Matt Alexander's lab

 0     
2 months ago



  **N** **NBI\_standard\_RNA-seq\_pipeline\_PE** Maintainer



One out of two RNA-seq pipelines offered by the NBI. Its name contain "standard" ...

 2     
2 months ago



  **N** **NBI\_quasi\_mapping\_RNA-seq\_pipeline\_PE** Maintainer



One out of two RNA-seq pipelines offered by the NBI. This pipeline utilizes the pse...

 1     
2 months ago



  **N** **NBI\_MeDIP-seq\_SE\_pipeline**



MeDIP-seq single-end pipeline

 1     
2 months ago



  **N** **NBI\_RRBS\_SE\_pipeline** Maintainer

RRBS single-end pipeline.

 1     
2 months ago

  **N** **NBI\_WGBS\_RRBS\_PE\_pipeline** Maintainer

WGBS and RRBS paired-end pipeline.

 1     
2 months ago

**GitLab “groups” are ideal for individual labs/cores**

- Unlimited number of projects (there is a limit in your own profile)
- Easy to collaborate/share

N

# NBI\_standard\_RNA-seq\_pipeline\_PE

Project ID: 446 | [Leave project](#)

★ Unstar

2

🔗 Fork

1

Clone ▾

[Add license](#) [40 Commits](#) [1 Branch](#) [14 Tags](#) [543 KB Files](#)

One out of two RNA-seq pipelines offered by the NBI. Its name contain "standard" as it uses the common alignment method (STAR), rather than pseudoaligners. This is the PE version of the standard pipeline.



## Auto DevOps

It will automatically build, test, and deploy your application based on a predefined CI/CD configuration.

Learn more in the [Auto DevOps documentation](#)

[Enable in settings](#)

master ▾

NBI\_standard\_RNA-seq\_pipeline\_PE / + ▾

History

🔍 Find file

Web IDE

**updated samtools=1.9**

Lara lanov authored 3 months ago

7d2f8039



SHA-1 hash is essential to git's method of version control

Access can be given to:

1. Specific user(s)
2. Everyone at UAB
3. Globally

They may also open "issues" with requests to add specific features etc.

# Project members

You can invite a new member to **NBI\_standard\_RNA-seq\_pipeline\_PE** or invite another group.

Invite member	Invite group
<div>Select members to invite</div> <div><div>Search for members to update or invite</div></div>	
<div>Choose a role permission</div> <div><div>Guest</div><div></div></div>	
<div><a href="#">Read more</a> about role permissions</div>	
<div>Access expiration date</div> <div><div>Expiration date</div></div>	
<div>Add to project</div>	<div>Import</div>

See more member permissions at: <https://docs.gitlab.com/ee/user/permissions.html>



Action	Guest	Reporter	Developer	Maintainer	Owner
Create new issue	✓ <sup>1</sup>	✓	✓	✓	✓
Create confidential issue	✓ <sup>1</sup>	✓	✓	✓	✓
View confidential issues	(✓) <sup>2</sup>	✓	✓	✓	✓
Leave comments	✓ <sup>1</sup>	✓	✓	✓	✓
See related issues	✓	✓	✓	✓	✓
See a list of jobs	✓ <sup>3</sup>	✓	✓	✓	✓
See a job log	✓ <sup>3</sup>	✓	✓	✓	✓
Download and browse job artifacts	✓ <sup>3</sup>	✓	✓	✓	✓
View wiki pages	✓ <sup>1</sup>	✓	✓	✓	✓
View license management reports ⓘ	✓ <sup>1</sup>	✓	✓	✓	✓
View Security reports ⓘ	✓ <sup>1</sup>	✓	✓	✓	✓
Pull project code	1	✓	✓	✓	✓
Download project	1	✓	✓	✓	✓
Assign issues		✓	✓	✓	✓
Assign merge requests			✓	✓	✓
Label issues		✓	✓	✓	✓
Label merge requests			✓	✓	✓
Create code snippets		✓	✓	✓	✓
Manage issue tracker		✓	✓	✓	✓
Manage labels		✓	✓	✓	✓

# GitLab – Starting a repository

- From GitLab:

## Projects

Your projects **Starred projects** Explore projects

Filter by name...

Last updated

New project

N

CIRC\_NBI / NBI\_standard\_RNA-seq\_pipeline Owner

One out of two RNA-seq pipelines offered by the NBI. Its name contain "standard" as it uses the common alignment method (STAR), rather than pseudoaligners.

★ 1 🔒

updated 1 day ago

G

Lara Ianov / General\_HPC\_setup Maintainer

Useful alias to be added to .bashrc in HPC

★ 1 🔒

updated 1 month ago

N

CIRC\_NBI / NBI\_WGBS\_RRBS\_PE\_pipeline Owner

WGBS and RRBS paired-end pipeline.

★ 1 🔒

updated 2 months ago

Blank project

Create from template

Import project

**Project name**

My awesome project

**Project URL**

https://gitlab.rc.uab.edu/

lianov



**Project slug**

my-awesome-project

Want to house several dependent projects under the same namespace? [Create a group.](#)

**Project description (optional)**

Description format

Path may be user profile or specific groups (e.g.: "CIRC\_NBI")

**Visibility Level**

- ☒ **Private**  
Project access must be granted explicitly to each user.
- ☐ **Internal**  
The project can be accessed by any logged in user.
- ☐ **Public**  
The project can be accessed without any authentication.

☐ **Initialize repository with a README**

Allows you to immediately clone this project's repository. Skip this if you plan to push up an existing repository.

Create project

Cancel

- From the command line:
  - Very nice when uploading a directory of existing files

#### Command line instructions

##### Git global setup

```
git config --global user.name "Lara Ianov"  
git config --global user.email "lianov@uab.edu"
```

##### Create a new repository

```
git clone git@gitlab.rc.uab.edu:lianov/gitlab_demo.git  
cd gitlab_demo  
touch README.md  
git add README.md  
git commit -m "add README"  
git push -u origin master
```

Or type a one-liner to stage and commit all **tracked** files:  
`git commit -a -m "message goes here"`

##### Existing folder

```
cd existing_folder  
git init  
git remote add origin git@gitlab.rc.uab.edu:lianov/gitlab_demo.git  
git add .  
git commit -m "Initial commit"  
git push -u origin master
```

Commands to upload files when initial project (in this case named "gitlab\_demo") was created in GitLab

##### Existing Git repository

```
cd existing_repo  
git remote rename origin old-origin  
git remote add origin git@gitlab.rc.uab.edu:lianov/gitlab_demo.git  
git push -u origin --all  
git push -u origin --tags
```



- From the command line:
  - Or create the project directly from the command line and push all files to the project

```
cd existing_folder
git init
git add .
git commit -m "initial commit message"
```

# With SSH

```
git push -u git@gitlab.rc.uab.edu:<namespace>/<project_name>.git master
git remote add origin git@gitlab.rc.uab.edu:<namespace>/<project_name>.git
```

# OR with HTTPS

```
git push -u https://gitlab.rc.uab.edu/<namespace>/<project_name>.git master
git remote add origin https://gitlab.rc.uab.edu/<namespace>/<project_name>.git
```

FYI on setting up SSH for GitLab: <https://docs.gitlab.com/ee/ssh/README.html>







# GitLab – Edit and commit

- GitLab contains an integrated development environment (IDE) – OK for small edits, but I do not use it for large projects/changes.

CIRC\_NBI > gitlab\_demo > Repository

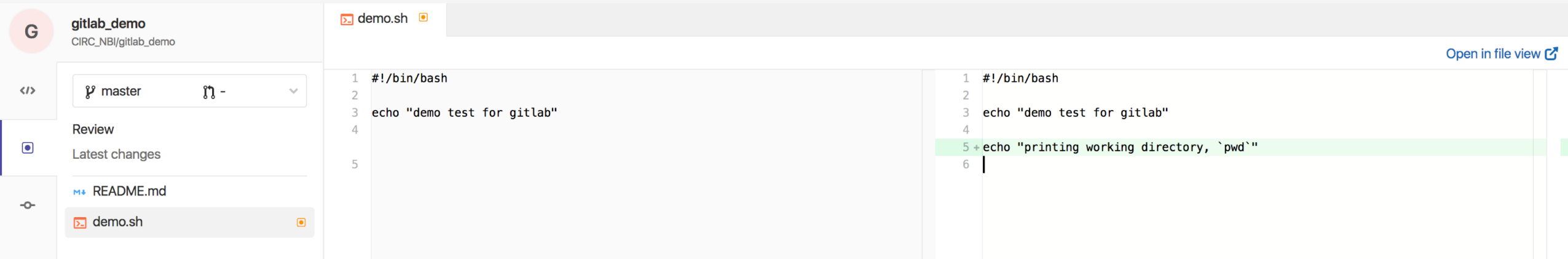
master gitlab\_demo / demo.sh Find file Blame History Permalink

 initial commit  
Lara lanov authored 6 minutes ago fb8a78fc 

 demo.sh 42 Bytes  Edit Web IDE Replace Delete

```
1 #!/bin/bash
2
3 echo "demo test for gitlab"
4
```

# GitLab's IDE

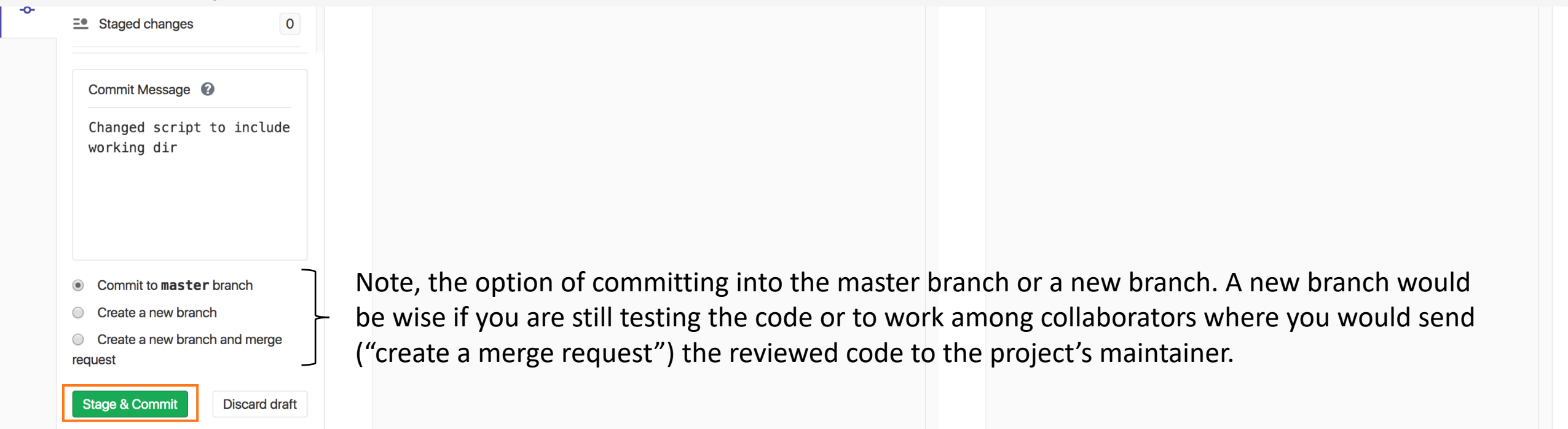


The screenshot shows the GitLab IDE interface. On the left, there's a sidebar with a user profile 'gitlab\_demo' and a file explorer showing 'demo.sh' selected. The main editor area displays a shell script 'demo.sh' with the following content:

```
1 #!/bin/bash
2
3 echo "demo test for gitlab"
4
5 + echo "printing working directory, `pwd`"
6
```

The new line of code is highlighted in green. A link 'Open in file view' is visible in the top right corner.

Hit “Stage and Commit” and include a message. Same as `git add . && git commit -m “message”`  
No need to push as this is being done directly in GitLab.



The screenshot shows the 'Stage and Commit' dialog box in the GitLab IDE. The dialog has a 'Commit Message' field with the text 'Changed script to include working dir'. Below the message field, there are three radio button options:

- ☒ Commit to **master** branch
- ☐ Create a new branch
- ☐ Create a new branch and merge request

The 'Stage & Commit' button is highlighted with a green border. A 'Discard draft' button is also visible. A bracket on the right side of the dialog points to the radio button options.

Note, the option of committing into the master branch or a new branch. A new branch would be wise if you are still testing the code or to work among collaborators where you would send (“create a merge request”) the reviewed code to the project’s maintainer.

# GitLab – Edit and commit

- ✓ GitLab contains an integrated development environment (IDE) – OK for small edits, but I do not use it for large projects/changes.
- Cloning the repo to your local computer and using your favorite editor is usually best for large changes.
  - `git clone <SSH or HTTPS>`

The screenshot shows the GitLab interface for a repository named 'gitlab\_demo' (Project ID: 649). The repository has 11 commits, 2 branches, 0 tags, and 14.2 MB of files. The 'Auto DevOps' feature is highlighted with a green checkmark and a button to 'Enable in settings'. The 'Clone' button is highlighted with an orange box and an arrow pointing to it with the text 'Click "Clone"'. The 'Clone with SSH' and 'Clone with HTTPS' options are shown, with the SSH URL 'git@gitlab.rc.uab.edu:lianov' and the HTTPS URL 'https://gitlab.rc.uab.edu/li' both highlighted with orange boxes and an arrow pointing to them with the text 'Copy SSH/HTTPS here'. The bottom of the page shows the 'master' branch selected, the repository name 'gitlab\_demo', and buttons for 'History', 'Find file', 'Web IDE', and a 'Clone' icon.

Click "Clone"

Copy SSH/HTTPS here

You may also download

# GitLab – Edit and commit

- ✓ GitLab contains an integrated development environment (IDE) – OK for small edits, but I do not use it for large projects/changes.
- Cloning the repo to your local computer and using your favorite editor is usually best for large changes.
  - `git clone <SSH or HTTPS>`

```
demo.sh
1  #!/bin/bash~
2  ~
3  echo "demo test for gitlab"~
4  ~
5  echo "printing working directory, `pwd`"~
6  ~
7  echo "line added locally after cloning, and then committed and pushed to GitLab"~
8  ~
```

Line added in my editor of choice in my computer and then:  
`git add .`  
`git commit -m "edit made locally in my editor"`  
`git push -u origin master`

# GitLab – Atom integration



- Atom is a great editor for both GitHub and GitLab as it integrates the git commands, allowing you to perform tasks such as `commit` and `push` directly from the editor. Explore Atom's packages such as [this](#) to learn more

```
1 #!/bin/bash
2
3 echo "demo test for gitlab"
4
5 echo "printing working directory, `pwd`"
6
7 echo "line added locally after cloning, and then committed and pushed to GitLab"
8
9 # A good FYI about the Atom editor:-
10 # changes below were made in Atom, staged, committed and pushed directly from Atom
11 echo "printing current date and time"
12 date
13
```

Highlights changes

Unstaged Changes

demo.sh

Stage All

Staged Changes

No changes

Changes made in Atom and committed etc. directly

Commit to master

Update README.md

edit made locally in my editor

Changed script to include working dir

push icon is available after committing → master ↑ Push 1

# GitLab – History

- Regardless of where changes were made (in GitLab IDE or locally and then pushed), all changes are easily tracked in GitLab:

master ▾


gitlab\_demo / + ▾

History


🔍 Find file



Web IDE

🔗 ▾



**edit made locally in my editor**  
Lara lanov authored 17 minutes ago

0fd77486 

Name	Last commit	Last update
 README.md	initial commit	1 hour ago
 demo.sh	edit made locally in my editor	17 minutes ago

# FYI command line version: `git log`



master

gitlab\_demo

Filter by commit message

28 Nov, 2018 3 commits



**edit made locally in my editor**

Lara lanov authored 25 minutes ago



**Changed script to include working dir**

Lara lanov authored 1 hour ago



**initial commit**

Lara lanov authored 2 hours ago

Note the changes in SHA-1

0fd77486



0917c098



fb8a78fc



edit made locally in my editor

FYI command line version: `git show <SHA-1>`

parent 0917c098 master

No related merge requests found

Showing 1 changed file with 2 additions and 0 deletions

Hide whitespace changes

Inline

Side-by-side

demo.sh 100644 → 100755

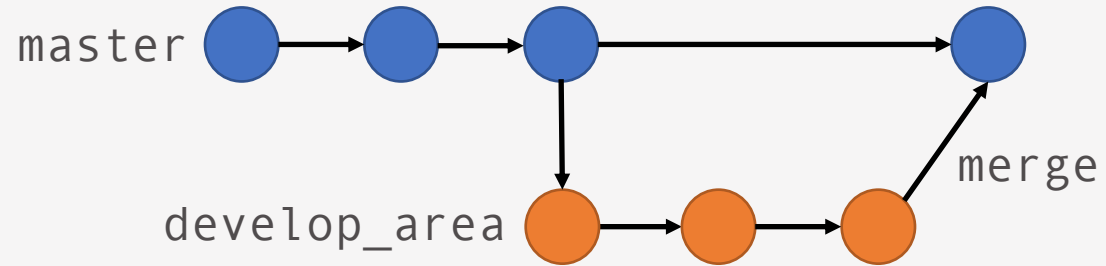


View file @ 0fd77486

```
... @@ -3,3 +3,5 @@
3  echo "demo test for gitlab"
4
5  echo "printing working directory, `pwd`"
```

```
... @@ -3,3 +3,5 @@
3  echo "demo test for gitlab"
4
5  echo "printing working directory, `pwd`"
6  +
7  + echo "line added locally after cloning, and then committed and pushed to GitLab"
```

# GitLab - Branch



- By default, all projects start with one branch, called `master`. All commits I have done in this demo so far have been made to `master`.
- But what if you would like to work on the repo without committing to the main branch? Maybe testing code without overwriting the `master` branch?
  - This is where `branching` is very useful, especially when more than one person is working on the same code.
  - In the case of more than one person, a collaborator can branch, edit independently at the isolated branch and when finalized, send a `merge` request to the maintainer who can approve to merge all changes into the `master` branch.





**gitlab\_demo**

Project ID: 936



☆ Star

0

🍴 Fork

0

Clone

Add license

1 Commit

1 Branch

0 Tags

2.1 MB Files

Lara Ianov > gitlab\_demo > **Branches**

**Overview**

Active

Stale

All

Filter by branch name

Delete merged branches

New branch

Protected branches can be managed in [project settings](#).

Active branches

**master** default protected

25e27b03 · Changes made in Atom and committed etc. directly · 23 hours ago



## New Branch

Branch name

Create from

Existing branch name, tag, or commit SHA

Create branch

Cancel



**gitlab\_demo**

Project ID: 649



Star

0

Fork

0

Clone

Add license

11 Commits

2 Branches

0 Tags

14.2 MB Files

You can now edit the code of the second branch and click on “branches” to compare and add a merge request

Overview   Active   Stale   All









Filter by branch name


Delete merged branches

New branch

Protected branches can be managed in [project settings](#).

Active branches

 <b>develop_area</b>					
 <a href="#">caeed48f</a> · Update demo.sh · 2 minutes ago					
		0   1	Merge request	Compare	 ▼ 
 <b>master</b> <span>default</span> <span>protected</span>					
 <a href="#">25e27b03</a> · Changes made in Atom and committed etc. directly · 23 hours ago					
					 ▼ 



Source

develop\_area

...

Target

master

Compare

Create merge request

Commits (1)



Update demo.sh

Lara lanov authored 3 minutes ago

caeed48f



Showing 1 changed file ▾ with 3 additions and 0 deletions

Hide whitespace changes

Inline

Side-by-side

demo.sh



View file @ caeed48f

...	@@ -10,3 +10,5 @@ echo "line added locally after cloning, and then committed and pushed to GitLab"	...	@@ -10,3 +10,5 @@ echo "line added locally after cloning, and then committed and pushed to GitLab"
10	# changes below were made in Atom, staged, committed and pushed directly from Atom	10	# changes below were made in Atom, staged, committed and pushed directly from Atom
11	echo "printing current date and time"	11	echo "printing current date and time"
12	date	12	date
		13	+
		14	+ #----- This represents finalized code ready to be merged -----
			\ No newline at end of file



# GitLab - Branch

- Again, using GitLab's IDE may be OK for small changes but I prefer local for larger changes:

```
git clone <SSH or HTTPS> # or git pull
cd <cloned_dir> # only needed if cloned
git branch -a
git checkout <branch_name> (switches the branch)
### modify code ###
git commit -am "message goes here" (adding in one-liner)
git push -u origin <branch_name>
```

- Atom's packages offer good features to do this as well

# Thanks!

A big thank you to UAB Research Computing for hosting GitLab and for taking the time to provide additional documentation and training

My contact info:

lianov@uab.edu

CIRC 252C | P: 205-996-5871

[NBI](#)