

GitLab: Getting Started



Table of Contents

| | |
|---|----|
| GitLab: Getting Started | 1 |
| Search/Filter Projects, Groups, & Subgroups | 4 |
| To search for projects, groups, & subgroups: | 4 |
| To filter projects and groups: | 4 |
| Forking & Merge Requests | 6 |
| To fork a repository: | 6 |
| To update forked project: | 7 |
| To create a merge request: | 8 |
| To review merge request: | 12 |
| To reject merge request: | 14 |
| To accept merge request: | 14 |
| Revert Commits | 16 |
| To revert a commit: | 16 |
| Transfer Projects to Users | 19 |
| To add users for project access: | 19 |
| To transfer the project to user: | 22 |
| Remove Projects | 23 |
| To remove the project: | 23 |
| Download Copy of Project | 25 |
| To download the copy of the project onto your computer: | 25 |
| Change Project Picture | 26 |
| To change project picture: | 26 |
| Using GitLab Editor | 28 |
| To add a new file to the project: | 28 |
| To edit existing files in the project: | 31 |
| Fixing Merge Conflicts | 34 |
| To fix merge conflicts: | 34 |
| Create Groups & Subgroups | 38 |
| To create a group: | 38 |
| To create a subgroup: | 39 |

| | |
|---|-------------------------------------|
| Add/Remove Users from Groups and Subgroups | 42 |
| To add users in the group:..... | 42 |
| To add users in the subgroup:..... | 43 |
| To remove users in the group: | 46 |
| To remove users in the subgroup:..... | 47 |
| Change User's Permission Levels in Groups/Subgroups | 49 |
| To change user permission levels in groups:..... | 49 |
| To change user permission levels in subgroups: | 51 |
| Transfer Projects to Groups/ Subgroups | 54 |
| To transfer projects to groups/subgroups: | 54 |
| Share Projects with Group/Subgroup | 57 |
| To share project with group: | 57 |
| Create Issues | 60 |
| To create issues:..... | 60 |
| Create Issue Labels..... | 63 |
| To create issue labels: | 63 |
| Using Issues Board/List | 66 |
| To add a new category in the issue board: | 66 |
| To move the category around in the issue board: | 69 |
| To remove the category in issue board:..... | 70 |
| User Settings | 73 |
| To change profile picture: | 73 |
| To change GitLab appearance:..... | 74 |
| Logging into GitLab..... | 77 |
| To log into GitLab: | 77 |
| Other Mentions..... | 80 |
| Permission Levels | 80 |
| Application Integrations | 81 |
| Administrator Settings (Administrator Only) | Error! Bookmark not defined. |

Search/Filter Projects, Groups, & Subgroups

GitLab allows users to search for groups, subgroups, and projects quickly and efficiency. Users can search for group names or titles that either begins, contains, and/or ends with the search input. Filtering narrows down the current list that matches with the user's input. Users can filter that either begins, contains, and/or ends with for groups, subgroups, projects, and/or user's last name.

To search for projects, groups, & subgroups:

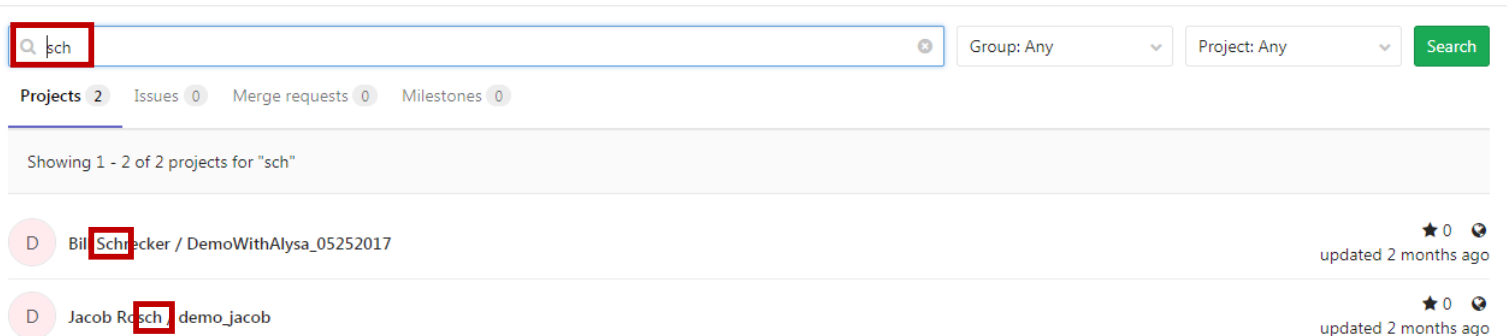
- 1) Located the text field at the top right header with a **microscope** icon and the word **Search** displayed in it.



- 2) Type in the **name/substring** of either groups, subgroups, and/or projects.
- 3) Press **Enter** on your keyboard to search for the input. This will take you to the search result page that will list all of the projects that matches with the input. Otherwise, the page states that there are no projects that matches with the search input.

Note: You can only search for groups/subgroups/projects that are only set to public visibility.

Search



To filter projects and groups:

- 1) On the top menu header, click on **Projects** or **Groups** tab.



- 2) Click on **Explore projects** or **Explore public groups** tab.

Projects New project

Your projects Starred projects **Explore projects** Filter by name... Last updated ▾

Groups New group

Your groups **Explore public groups** Filter by name... Last created ▾

OR

3) (For Projects Only) Click on the **All** tab.

Projects New project

Your projects Starred projects **Explore projects** Filter by name... Last updated ▾

Trending Most stars **All** Visibility: Any ▾

4) Type into the text field that displayed **Filter by name...** to show those that matches the input.
The list will auto-refresh as users continue to type or remove letters.

Projects New project

Your projects Starred projects **Explore projects** **Filter by name...** Last updated ▾

Trending Most stars **All** Visibility: Any ▾

Forking & Merge Requests

Forking and merge request goes hand-in-hand and are considered to be very important in developing products or improving projects. Forking allows users to create their copy of the master project under their name and implement their changes while testing for errors. Once users are ready to incorporate their changes into the master project, users will create merge requests that allows other users to check and review new changes. Users are encouraged to have a discussion about the changes before agreeing to accept or reject until new improvements.

To fork a repository:

- 1) Click on the **project name** that you want to work on.
- 2) Underneath the project image and project name, click on the **Fork** button.

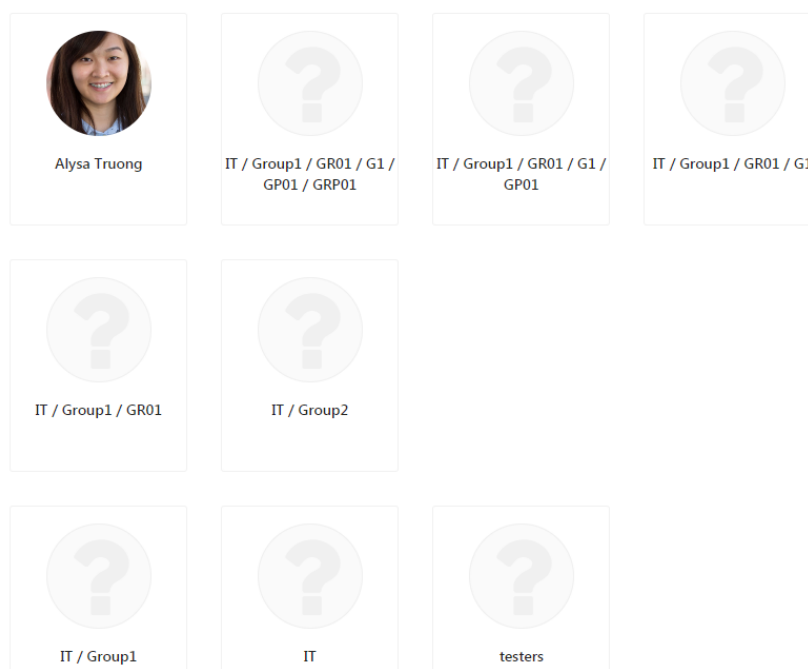


- 3) Select a **user/group/subgroup** to fork the project.

Fork project

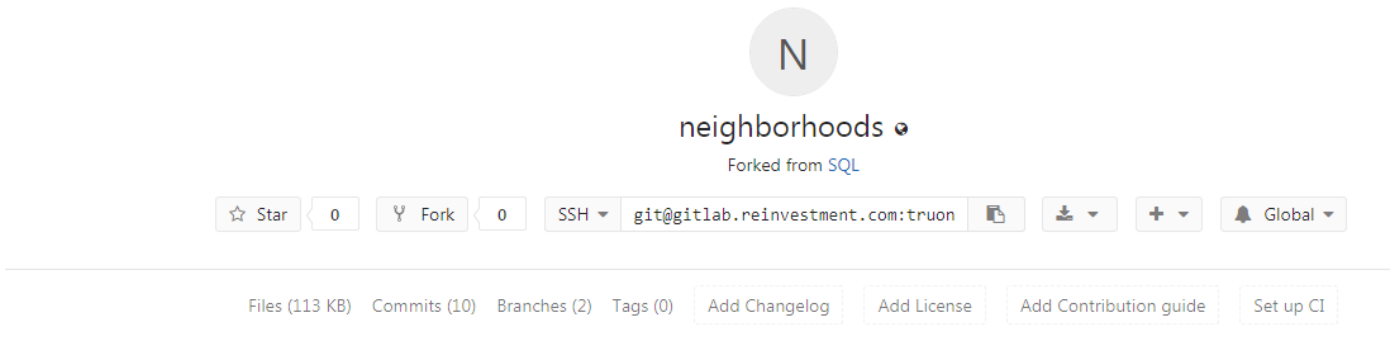
A fork is a copy of a project.
Forking a repository allows you to make changes without affecting the original project.

Click to fork the project to a user or group



- 4) You should get a **message** displayed across the screen, stating that the project was successfully forked. Underneath the project name, you should see that it is **forked from**

The project was successfully forked.



To update forked project:

Note: You may need to use *Git Bash* to pull the latest changes into your forked project repository. However, there is another simpler way which is to delete your current fork repository and then go back to the master project to create a new fork there.

- 1) Open **Git Bash**.
- 2) Clone your **forked repository** with the following command line:
 - a. `git clone http://gitlab.reinvestment.com/../../.git`

```
truonga@D_4JQRG42 MINGW64 ~/Desktop
$ git clone http://gitlab.reinvestment.com/truonga/neighborhoods.git
Cloning into 'neighborhoods'...
remote: Counting objects: 24, done.
remote: Compressing objects: 100% (17/17), done.
remote: Total 24 (delta 5), reused 21 (delta 5)
Receiving objects: 100% (24/24), done.
Resolving deltas: 100% (5/5), done.
```

Note: This remote will be called *origin* as default.

- 3) Move into that project folder in *Git Bash* with the following command:
 - a. `cd [project folder name]`

```
truonga@D_4JQRG42 MINGW64 ~/Desktop
$ cd neighborhoods
```

- 4) Enter the following command line:

- a. `git remote add upstream http://gitlab.reinvestment.com/../../.git`

```
truonga@D_4JQRG42 MINGW64 ~/Desktop/neighborhoods (master)
$ git remote add upstream http://gitlab.reinvestment.com/testers/SQL/neighborhoods.git
```

Note: There is a difference between **origin** and **upstream** remotes. Consider **origin** as the place you would want to make changes to only while **upstream** is the place where you should only get the latest changes.

- 5) Enter the following command line:

- a. `git pull upstream master`

```
truonga@D_4JQRG42 MINGW64 ~/Desktop/neighborhoods (master)
$ git pull upstream master
remote: Counting objects: 14, done.
remote: Compressing objects: 100% (14/14), done.
remote: Total 14 (delta 2), reused 0 (delta 0)
Unpacking objects: 100% (14/14), done.
From gitlab.reinvestment.com:testers/SQL/neighborhoods
 * branch            master       -> FETCH_HEAD
 * [new branch]      master       -> upstream/master
Updating 0f99ab3..6d1809a
Fast-forward
 newFile | 4 +++-
 1 file changed, 3 insertions(+), 1 deletion(-)
```

- 6) Once it is successfully updated, enter the following command line:

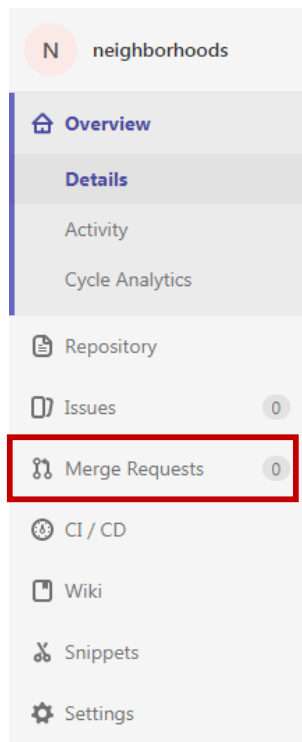
- a. `git push origin master`

```
truonga@D_4JQRG42 MINGW64 ~/Desktop/neighborhoods (master)
$ git push origin master
Counting objects: 14, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (14/14), done.
Writing objects: 100% (14/14), 1.74 KiB | 0 bytes/s, done.
Total 14 (delta 2), reused 0 (delta 0)
To gitlab.reinvestment.com:truonga/neighborhoods.git
 0f99ab3..6d1809a master -> master
```

To create a merge request:

- 1) Click on the **main project name**.

- 2) On the left side bar, click on **Merge Requests** tab. The number next to the Merge Requests tab represents the number of requests for that project.



- 3) Click on the **New merge request** button at the top right.



- 4) Check to see if the master project is under the **Target branch** section.



New Merge Request

Source branch

truonga/neighborhoods ▼ Select source branch ▼

Target branch





testers/SQL/neighborhoods ▼ master ▼

 officially deleted test file with space
Alysa Truong committed 2 weeks ago
528069ac 

Compare branches and continue





- 5) Under the Source branch, click on the **first drop-down** list to select the forked project that you made changes under.

New Merge Request

| Source branch | Target branch |
|---|--|
| <div>truonga/neighborhoods ▾</div> <div>master ▾</div> | <div>testers/SQL/neighborhoods ▾</div> <div>master ▾</div> |
| <div> Add new raining file Alysa Truong committed 2 minutes ago 0f99ab35 </div> | <div> officially deleted test file with space Alysa Truong committed 2 weeks ago 528069ac </div> |
| <div>Compare branches and continue</div> | |





- 6) Select the **branch** that you made changes under (If didn't add new branch, keep it as **master**).

New Merge Request

| Source branch | Target branch |
|---|--|
| <div>truonga/neighborhoods ▾</div> <div>master ▾</div> | <div>testers/SQL/neighborhoods ▾</div> <div>master ▾</div> |
| <div> Add new raining file Alysa Truong committed 2 minutes ago 0f99ab35 </div> | <div> officially deleted test file with space Alysa Truong committed 2 weeks ago 528069ac </div> |
| <div>Compare branches and continue</div> | |

- 7) Click on **Compare branches and continue** button. GitLab will find if there are any difference between two projects. If no difference between the two projects, GitLab displays an error message that there is none and to select another source.

New Merge Request

| Source branch | Target branch |
|---|--|
| <div>truonga/neighborhoods ▾</div> <div>master ▾</div> | <div>testers/SQL/neighborhoods ▾</div> <div>master ▾</div> |
| <div> Add new raining file Alysa Truong committed 2 minutes ago 0f99ab35 </div> | <div> officially deleted test file with space Alysa Truong committed 2 weeks ago 528069ac </div> |
| <div>Compare branches and continue</div> | |

- 8) Choose to create a **new merge request title** or **keep title** as it is. Users can decide to include **WIP:** (Work In Progress) status to the title, which prevents a request from being merged into the master project before it is ready to do so. To remove the WIP status, delete **WIP:** in the merge request title.

New Merge Request

From `truonga/neighborhoods:master` into `testers/SQL/neighborhoods:master`

Title

Remove the **WIP:** prefix from the title to allow this **Work In Progress** merge request to be merged when it's ready.

Add [description templates](#) to help your contributors communicate effectively!

- 9) Type in the **description** of the merge request that briefly explains the new changes.

Description

Write

Preview

B **I** **”** **</>** **≡** **≡** **✓** **✕**

Write a comment or drag your files here...

Markdown and slash commands are supported

Attach a file

- 10) Select a **user** who has access to the master project to be assigned to the merge request. That user will be responsible with reviewing the changes and oversee this current merge request.

Assignee

Assignee

▼

Assign to me

Milestone

Milestone

▼

Labels

Labels

▼

- 11) Select a **label** that describes the merge request (for organizing purposes).

Assignee

Assignee

▼

Assign to me

Milestone

Milestone

▼

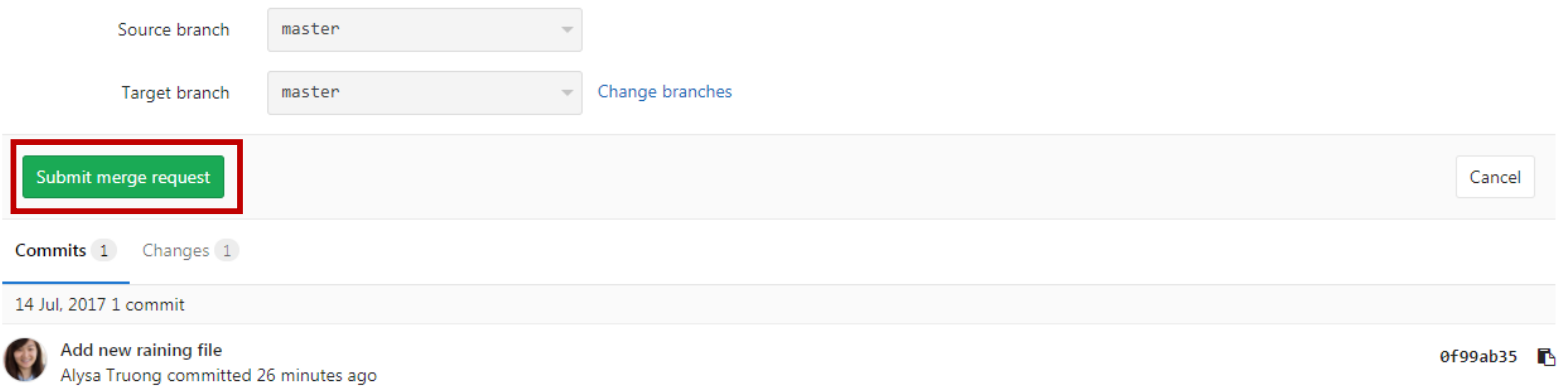
Labels

Labels

▼

Note: To create your own labels, follow the instructions under **Create Issues Labels** section in this guide.

- 12) Click on **Submit merge request** button to submit the request. This is notify the assigned user of a new merge request.




Source branch: master

Target branch: master [Change branches](#)

Submit merge request [Cancel](#)

Commits 1 Changes 1

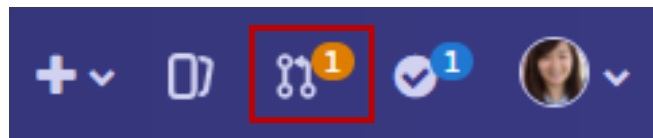
14 Jul, 2017 1 commit

 Add new raining file
Alysa Truong committed 26 minutes ago 0f99ab35

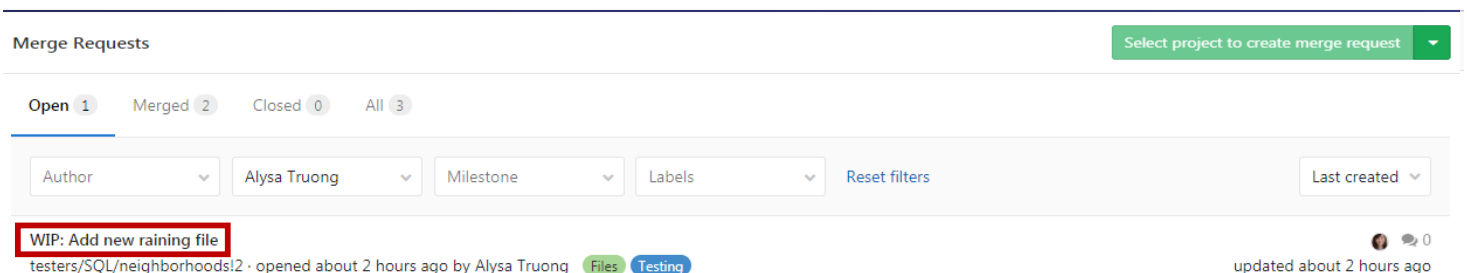
Note: The **Commit** tab shows all of the commits that are included in the merge request. The **Changes** tab displays all new changes within the forked project that are included in the merge request.

To review merge request:

- 1) On the right-top of the page, click on the **third icon** from the right. An orange icon with a number is shown, indicating the amount of merge requests that are open and pending.



- 2) The Merge Request page will list all of the requests that are open and pending submitted by you. Users can filter through multiple requests with the filter selections provided. Click on the **merge request name** to view the request in detail.



Merge Requests [Select project to create merge request](#)


Open 1 Merged 2 Closed 0 All 3

Author: Alysa Truong Milestone: Labels: [Reset filters](#) Last created:

WIP: Add new raining file

testers/SQL/neighborhoods12 · opened about 2 hours ago by Alysa Truong [Files](#) [Testing](#) updated about 2 hours ago

- 3) Click on **Changes** tab to review changes in the files in greater details.

[Open](#) Merge request !2 opened about 3 hours ago by  Alysa Truong

[Close](#) [Edit](#)

WIP: Add new raining file

rain, rain go away come again another day

Edited 24 minutes ago by Alysa Truong

Request to merge `truonga:master` into `master`

[Check out branch](#)



[Merge](#) This merge request is currently Work In Progress and therefore unable to merge [Resolve WIP status](#)

You can merge this merge request manually using the [command line](#).



Discussion 0

Commits 1

Changes 1

Showing 1 changed file with 4 additions and 0 deletions

[Hide whitespace changes](#)

[Inline](#)

[Side-by-side](#)

newFile 0 → 100644



[Edit](#)

[View file @ 0f99ab35](#)

```
1 + it's raining
```

```
2 +
```

```
3 + it's pouring
```

```
\ No newline at end of file
```

- 4) Click on **Discussion** tab to begin discussing about the request (what is good, what needs to be changed, etc.). Click on **Comment** button to post your comment. This will notify the request creator of a new comment made to the merge request.



Discussion 0

Commits 1

Changes 1

[Write](#) [Preview](#)

[B](#) [I](#) ["](#) [</>](#) [≡](#) [≡](#) [✓](#) [✕](#)

Write a comment or drag your files here...

Markdown and slash commands are supported

[Attach a file](#)

[Comment](#)

[Close merge request](#)



REINVESTMENT
FUND

To reject merge request:

- 1) Click on **Close merge request** button.



To accept merge request:

- 1) (For **WIP**) Click on **Resolve WIP status** button to remove the **WIP:** from the merge request title when it is ready to be merged.

Open Merge request !2 opened about 3 hours ago by Alysa Truong Close Edit

WIP: Add new raining file


rain, rain go away come again another day

Edited 22 minutes ago by Alysa Truong

Request to merge `truonga:master` into `master` Check out branch

Merge This merge request is currently Work In Progress and therefore unable to merge Resolve WIP status

You can merge this merge request manually using the [command line](#).



The merge request can now be merged.

Open Merge request !2 opened about 3 hours ago by Alysa Truong Close Edit

Add new raining file

rain, rain go away come again another day

Edited 34 minutes ago by Alysa Truong

Request to merge `truonga:master` into `master` Check out branch

Merge ☐ Remove source branch Modify commit message

You can merge this merge request manually using the [command line](#).

- 2) Click on **Merge** button to allow the commit to merge into master project.

Request to merge `truonga:master` into `master`

Check out branch



Merge

☐ Remove source branch

Modify commit message

You can merge this merge request manually using the [command line](#).



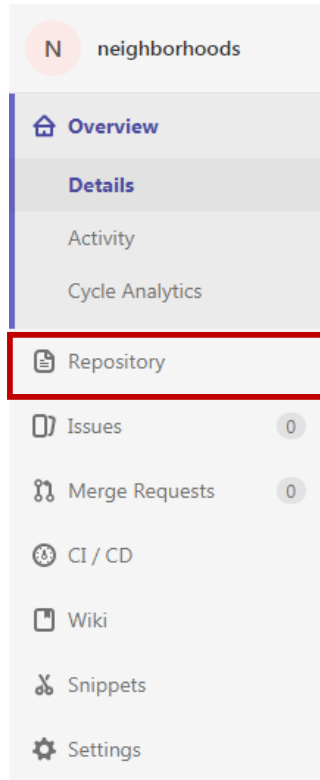
REINVESTMENT
FUND

Revert Commits

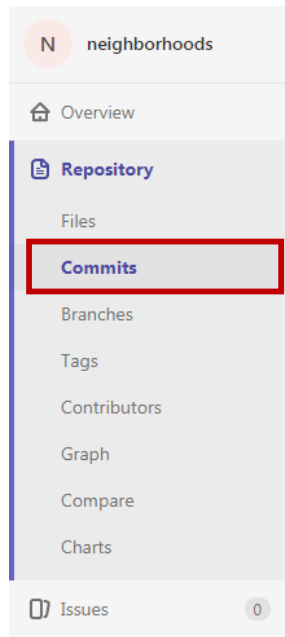
In any case of accidents, errors, etc. done the master project, personal, or forked repository, users can easily turn the project back to the previous state or version within a few steps by using the GitLab web interface.

To revert a commit:

- 1) Click on the **project name**.
- 2) Click on **Repository** tab on the left side of the page.



- 3) Click on **Commits** tab under the top header. This will display all of the pushed commits of the selected repository.



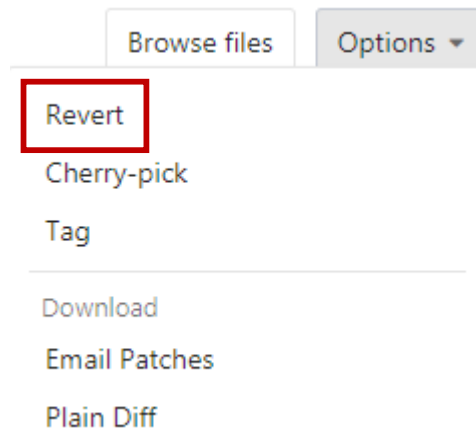
- 4) Click on the **commit** you wish to revert.
- 5) Click on **Options** at the top right-hand side of the page

Commit 0f99ab35 authored about 4 hours ago by Alysa Truong

Browse files

Options ▾

- 6) Select the **Revert** option.



- 7) Keep the default settings and click on **Revert** button. This will redirect you to back to the project home page.

Revert this commit

Revert in branch

master

☒ Start a **new merge request** with these changes

Revert

Cancel

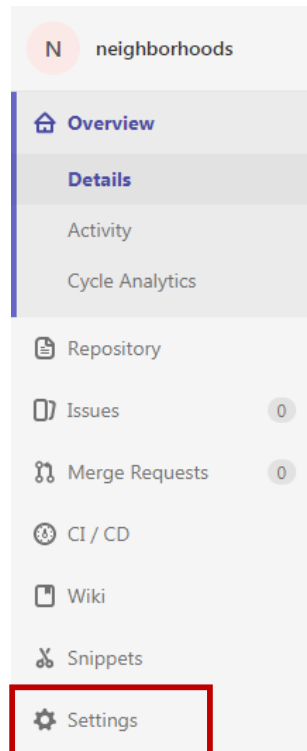
- 8) You should see a new commit on the project home page, starting with ***Merge branch 'revert... into master.'*** Clicking on the **Commits** tab will show other commits with ***Revert [commit message]***.

Transfer Projects to Users

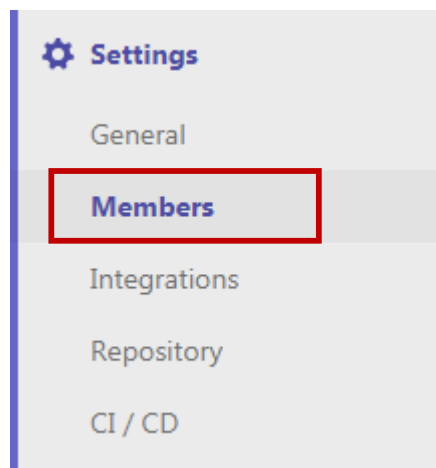
Transferring projects to another user requires that user to have access to the project (whether it is personal or group repository). Once the project has been transferred to the new user, the project will no longer be hold under the current/former owner since there is a new ownership of the repository.

To add users for project access:

- 1) Click on your **project name**.
- 2) Click on **Settings** at the left-side bar.



- 3) Click on the **Members** tab.



- 4) Under the **Add member** section, type in the user's name. A list will displayed below the input field and click on the user name once you locate it.

Project members

You can add a new member to **neighborhoods** or share it with another group.

| Add member | Share with group |
|---|------------------|
| <div>Select members to invite</div> <div>Search for members to update or invite</div> <div>Choose a role permission</div> <div>Guest</div> <div>Read more about role permissions</div> <div>Access expiration date</div> <div>Expiration date</div> <div><div>Add to project</div><div>Import</div></div> | |

- 5) Select the **permission level** for the user.

Project members

You can add a new member to **neighborhoods** or share it with another group.

| Add member | Share with group |
|---|-------------------------|
| Select members to invite | |
| <input type="text" value="Search for members to update or invite"/> | |
| Choose a role permission | |
| <div>Guest</div> | |
| Read more about role permissions | |
| Access expiration date | |
| <input type="text" value="Expiration date"/> | |
| <button>Add to project</button> | <button>Import</button> |

6) Click on the **Add to project** button.




Project members

You can add a new member to **neighborhoods** or share it with another group.

| Add member | Share with group |
|---|-------------------|
| Select members to invite | |
| <input type="text" value="Search for members to update or invite"/> | |
| Choose a role permission | |
| <div>Guest</div> | |
| Read more about role permissions | |
| Access expiration date | |
| <input type="text" value="Expiration date"/> | |
| <div><div>Add to project</div></div> | <div>Import</div> |

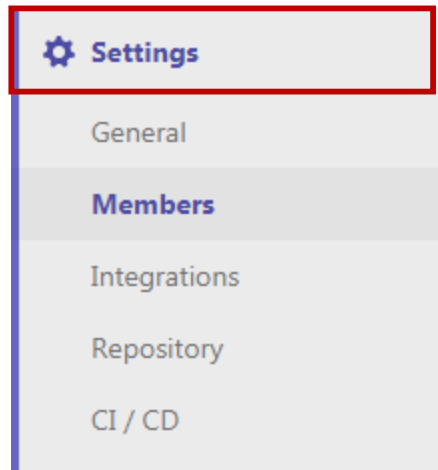
7) You should see the newly added user's name under the **Existing members and groups** section.

Existing members and groups

| Members of neighborhoods 3 | | | Find existing members by name | Name, ascending |
|---|--|--------------------|-------------------------------|-----------------|
|  | Administrator @root · testers / SQL | Joined a month ago | | Owner |
|  | Alysa Truong @truong · It's you · testers | Joined a year ago | | Master |
|  | VM-test @VMtest · testers / SQL | Joined a month ago | | Developer |

To transfer the project to user:

- 1) Click on the **project name**.
- 2) Click on **Settings** on the left side of the page.



- 3) Under the **Transfer project to new group**, click on the field to select the user you want to transfer the project.

Transfer project to new group

Please select the group you want to transfer this project to in the dropdown to the right.

Select a new namespace

- Be careful. Changing the project's namespace can have unintended side effects.
- You can only transfer the project to namespaces you manage.
- You will need to update your local repositories to point to the new location.
- Project visibility level will be changed to match namespace rules when transferring to a group.

Transfer project

- 4) Click on **Transfer project** button.

Transfer project to new group

Please select the group you want to transfer this project to in the dropdown to the right.

Select a new namespace

- Be careful. Changing the project's namespace can have unintended side effects.
- You can only transfer the project to namespaces you manage.
- You will need to update your local repositories to point to the new location.
- Project visibility level will be changed to match namespace rules when transferring to a group.

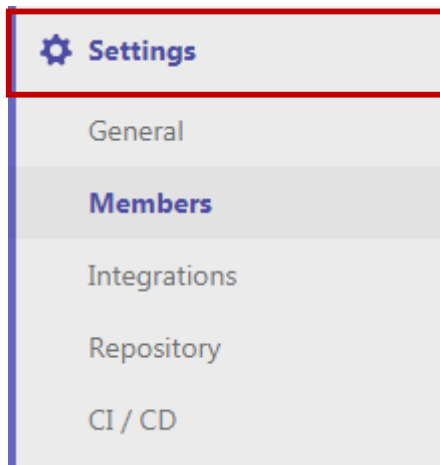
Transfer project

Remove Projects

In some cases, projects are no longer needed anymore. After (triple) confirming that projects are no longer needed in the future (whether to pick it up again or improvement for new projects), projects are removed permanently from the server with a click of a button (after a few clear warnings).

To remove the project:

- 1) Click on the **project name**.
- 2) On the top project header, click on **Settings**.



- 3) Scroll down to the bottom of the Setting page to find the Remove Project section. Click on the **Remove project** button.

Remove project

Removing the project will delete its repository and all related resources including issues, merge requests etc.

Removed projects cannot be restored!

Remove project

- 4) Type in the **project name** as indicated in the instruction into the text field provided.

Confirmation required

You are going to remove Alysa Truong / RStudio. Removed project CANNOT be restored! Are you ABSOLUTELY sure?

This action can lead to data loss. To prevent accidental actions we ask you to confirm your intention.

Please type **RStudio** to proceed or close this modal to cancel.

Confirm

- 5) Click on **Confirm** button. GitLab will immediately remove the project, depending on the current file size of the project.

Confirmation required



You are going to remove Alysa Truong / RStudio. Removed project CANNOT be restored! Are you ABSOLUTELY sure?

This action can lead to data loss. To prevent accidental actions we ask you to confirm your intention.
Please type **RStudio** to proceed or close this modal to cancel.

Confirm

Download Copy of Project

Whether in the case users would want a complete and current copy of the project offline, users can download a zip file of the chosen repository from the GitLab server. The project offers several compressed file extension of their choosing.

To download the copy of the project onto your computer:

- 1) Click on the **project name** you would like to download.
- 2) Click on the **downward arrow and with hard drive** icon button (hovering over the icon will show *Download*).

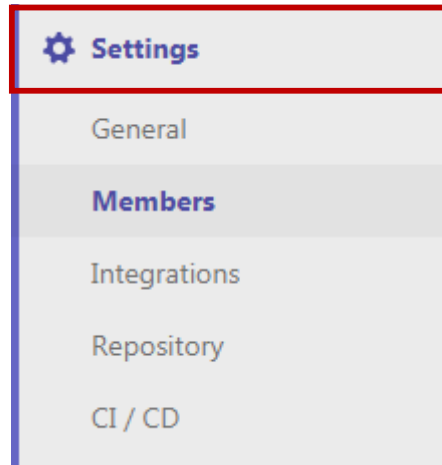


Change Project Picture

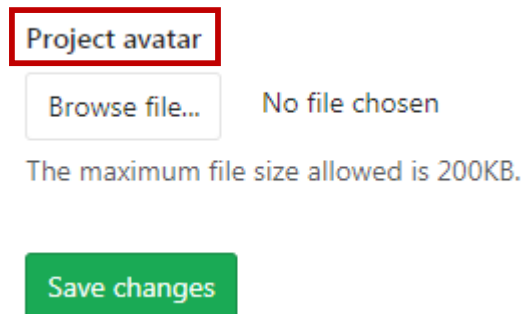
Sometimes, project needs something to make it stand out from other project repository, such as a project logo or something fun to represent the projects. With no project image, the default image for the project repository is the first letter of the project title.

To change project picture:

- 1) Click on the **project name**.
- 2) On the top project header, click on the **Settings** tab.



- 3) Scroll down to the **Project avatar** section in the *Project Settings* page.



- 4) Click on **Browse file...** button.

Project avatar

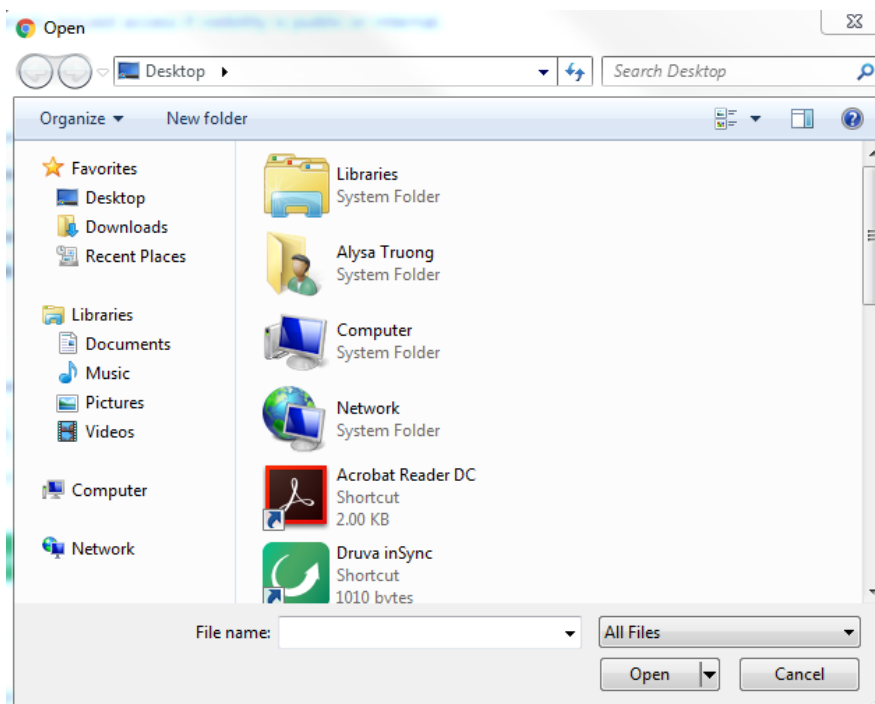
Browse file...

No file chosen

The maximum file size allowed is 200KB.

Save changes

- 5) Locate and choose the **image** from your computer.



- 6) Click on the **Save changes** button when complete.

Project avatar

Browse file...

No file chosen

The maximum file size allowed is 200KB.

Save changes

Using GitLab Editor

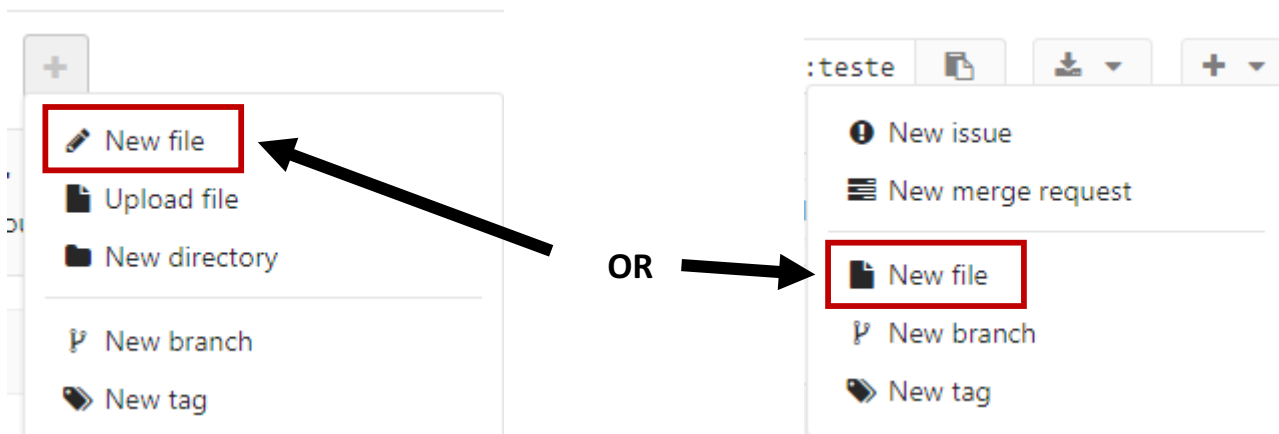
GitLab web interface provides a feature where users can make quick edits to files of the project or easily create a new text file in the project. The GitLab editor opens various file extensions and the contents are shown as it would when it is opened with a basic text editor available on the computer (may maintains its original style and format of the file).

To add a new file to the project:

- 1) Click on the **project name**.
- 2) Click on the **plus icon (+)** button. This will show several items that you can add to the project.





- 3) Select the **New file** option. This will take you to the web editor.



4) In the **File Name** text field, enter the *title* of the file.



New file | Template Choose type ▾

master / File name   Soft wrap | text ▾

1

Commit message Add new file

Target Branch master

5) Below the File name text field, create the **file contents**.

New file Template Choose type ▾

master / File name Soft wrap text ▾

Commit message Add new file

Target Branch master

Commit changes Cancel

- 6) Type your **commit** message. Click on **Commit changes** button to commit and push the commit to the project.

New file Template Choose type ▾

master / File name Soft wrap text ▾

1

Commit message Add new file

Target Branch master

Commit changes Cancel

To edit existing files in the project:

- 1) Click on the **project name** you want to edit.
- 2) Click on **Files** tab on the second project header.



neighborhoods

☆ Star 0

🍴 Fork 2

SSH ▾ git@gitlab.reinvestment.com:teste

📄

⬇ ▾

+ ▾

🔔 Global ▾

Files (215 KB)

Commits (12)

Branches (2)

Tags (0)

Add Changelog

Add License

Add Contribution guide

Set up CI

- 3) Click on the **file name** shown in the list. This will redirect users to another page where the contents of the file can be viewed in greater details.

master
neighborhoods /

Find file
History

Merge branch 'master' into 'master'
Alysa Truong committed 3 days ago
a1db4860

| Name | Last commit | Last Update |
|-------------------|--|-------------|
| BoxplotGraphics.R | add new R file and deleted old test file | 3 weeks ago |
| newFile | Add new raining file | 4 days ago |
| testFile.txt | Update testFile.txt | 4 weeks ago |

- 4) Right above the contents of the file on the right side of the page, click on **Edit** button. This will take you to the GitLab editor with the current file open to make changes.

master
neighborhoods / testFile.txt

Find file
Blame
History
Permalink

Update testFile.txt
VM-test committed 4 weeks ago
cacb688a

testFile.txt 133 Bytes

Edit
Replace
Delete

```

1  testing with text file first before moving onto VS solution files and RStudio files
2
3  Successfully edit the text file on GitLab editor

```

- 5) Edit the **file content** in the middle section of the page.

Edit file

Write

Preview changes

master

testFile.txt

Soft wrap

text

1

2

3

Testing with text file first before moving onto VS solution files and RStudio files

Successfully edit the text file on GitLab editor

Commit message

Update testFile.txt

Target Branch

master

Commit changes

Cancel

- 6) Once you completed editing the file, type the **commit** message.

Commit message

Update testFile.txt

Target Branch

master

Commit changes

- 7) Click on **Commit changes** button to commit and push the commit to the project.

Commit message

Update testFile.txt

Target Branch

master

Commit changes



Fixing Merge Conflicts

GitLab makes sure there are no merge conflicts, or problems, between two files of the master project and forked project. It checks for anything that might cause any issues in case that several other users had also worked on the same section of the files. This prevents any accident of overlapping user's work with another user and cause user to manually resolve it before merging into the master project. This saves a lot of time and frustrations from going back and determine what the previous solution.

Someone edited the file the same time you did. Please check out [the file](#) and make sure your changes will not unintentionally remove theirs.

Edit file

Write

Preview changes

master

newFile

Soft wrap

text

1

It's raining

2

3

it's pouring

4

5

nevermind... it's never gonna end!

To fix merge conflicts:

- 1) Open up a **merge request** that has a merge conflict.



- 2) (For **WIP**) Remove the **WIP** status from the merge request by clicking the **Edit** button and delete the **WIP:** from the merge request's title.

Open

Merge request !3 opened about a minute ago by Alysa Truong

Close

Edit

WIP: Update newFile with mister sun!

added two new lines for Mister Sun!

Request to merge truonga:master into master (1 commit behind)

Check out branch

Merge

There are merge conflicts.

Resolve conflicts

Merge locally

You can merge this merge request manually using the [command line](#).

- 3) Click on the **Resolve conflicts** button. This will redirect the user to viewing the conflicts of the file.

WIP: Update newFile with mister sun!

added two new lines for Mister Sun!

Request to merge `truonga:master` into `master` (1 commit behind)

Check out branch

Merge

There are merge conflicts.

Resolve conflicts

Merge locally

You can merge this merge request manually using the [command line](#).

4) Choose to follow **Step A** or **Step B** below:

Update newFile with mister sun!

added two new lines for Mister Sun!

Edited about a minute ago by Alysa Truong

Showing **1 conflicts** between `master` and `master`

Inline Side-by-side

newFile

Interactive mode Edit inline View file @40ca416

| | | | |
|---|---|--------------------------------|------------|
| 4 | 4 | @@ -4,8 +4,9 @@ it's pouring | |
| 5 | 5 | nevermind.. it is sunny today! | |
| 6 | 6 | | |
| | | HEAD//our changes | Use ours |
| 7 | | Oh Mister Sun, Sun | |
| 8 | | Mister golden sun! | |
| | | | |
| | 7 | //random spot saved | |
| | 8 | //another spot saved toooo | |
| | | origin//their changes | Use theirs |

Commit message

Merge branch 'master' into 'master'

Conflicts:

newFile

Commit conflict resolution

Cancel

- Select which **option** (*Use ours* as your change or *Use theirs* as someone's change) to keep in the file.
- Users can manually edit modify the files to resolve the merge conflict with the editor by clicking on the **Edit inline** button. Modify to your liking, type your new commit message.

Note: Make sure to remove the arrows (<<<< and >>>>) and lines (====) when manually editing the file on the GitLab editor (example provided below after Step 7).

- 5) Click on the **Commit conflict resolution** button to commit and push the change.

Commit message


```
Merge branch 'master' into 'master'
```

```
# Conflicts:  
# newFile
```

Commit conflict resolution Cancel

- 6) You should see a **blue banner** message that states all merge conflicts were resolved and that the merge request can be merged successfully.

All merge conflicts were resolved. The merge request can now be merged.

Open Merge request !3 opened 15 minutes ago by  Alysa Truong

Close

Edit

Update newFile with mister sun!

added two new lines for Mister Sun!

Edited 13 minutes ago by Alysa Truong

Request to merge `truonga:master` into `master`

Check out branch



Merge The source branch HEAD has recently changed. Please reload the page and review the changes before merging.

- 7) **Refresh** the page. You should be able to merge the changes successfully. Click on the **Merge** button to merge your changes to the master project.

Update newFile with mister sun!

added two new lines for Mister Sun!

Edited 17 minutes ago by Alysa Truong

Request to merge `truonga:master` into `master`

Check out branch



Merge

☐ Remove source branch

Modify commit message

You can merge this merge request manually using the [command line](#).

Note: Example image for manually editing the file on the GitLab editor to try to resolve the merge conflict.

Update newFile with mister sun!

added two new lines for Mister Sun!

Edited 3 minutes ago by Alysa Truong

Showing 1 conflicts between master and master

Inline Side-by-side

newFile

Interactive mode Edit inline View file @40ca416

1 it's raining
2
3 it's pouring
4
5 nevermind.. it is sunny today!
6
7 <<<<<< newFile
8 Oh Mister Sun, Sun
9 Mister golden sun!
10 =====
11 //random spot saved
12 //another spot saved toooo
13 >>>>>> newFile
14

Commit message

Merge branch 'master' into 'master'

Conflicts:
newFile

Commit conflict resolution

Cancel

Note: The GitLab editor might not be available for resolving conflicts if the file is over 200 KB in size, file is binary and not text, already contains conflict markers, and does not exist under the same path in both branches.

Note: For more information on resolving merge conflict using GitLab editor: read [this](#).

Create Groups & Subgroups

GitLab offers users to create groups and subgroups to help organize large projects as well as giving certain people access to necessary projects. Users can create as many groups and subgroups within groups; there is no limit. Users can also create a subgroup within another subgroup. However, GitLab can only hold a maximum of 21 nested groups (meaning there are a max of 21 levels or tiers in the group hierarchy).

To create a group:

- 1) Click on the **Groups** tab at the top header.



- 2) On the right side of the Group page, click on the **New group** button.



- 3) Fill in the **text fields** provided on the page.

New Group

Group path:

Group name:

Description:

Group avatar:
The maximum file size allowed is 200KB.

Visibility Level

☒ Private
The group and its projects can only be viewed by members.

☐ Internal
The group and any internal projects can be viewed by any logged in user.

☐ Public
The group and any public projects can be viewed without any authentication.

Some visibility level settings have been restricted by the administrator.

- A group is a collection of several projects
- Members of a group may only view projects they have permission to access
- Group project URLs are prefixed with the group namespace
- Existing projects may be moved into a group

Create group

Legend:



Required

ge 38 | 81

Optional

8/2017

- 4) Click on the **Create group** button when complete.

New Group

Group path

Group name

Description

Group avatar
The maximum file size allowed is 200KB.

Visibility Level

- ☒ **Private**
The group and its projects can only be viewed by members.
- ☐ **Internal**
The group and any internal projects can be viewed by any logged in user.
- ☐ **Public**
The group and any public projects can be viewed without any authentication.

Some visibility level settings have been restricted by the administrator.

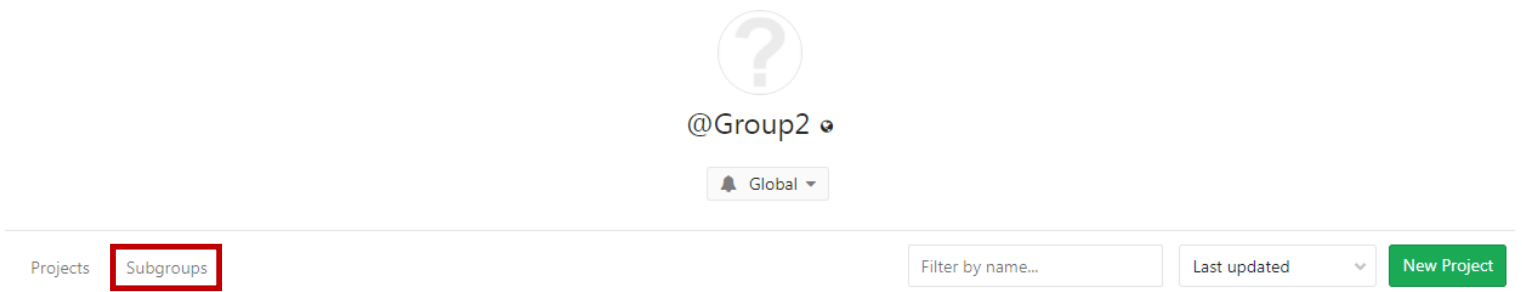
- A group is a collection of several projects
- Members of a group may only view projects they have permission to access
- Group project URLs are prefixed with the group namespace
- Existing projects may be moved into a group

To create a subgroup:

- 1) Click on the **Groups** tab at the top header.



- 2) From the list, click on the **group name** that you wish to create a subgroup.
- 3) On the group project page, click on **Subgroups** tab.



- 4) Click on the **New Subgroup** button.



@Group2

Global

Projects Subgroups

Filter by name

New Subgroup

5) Fill in the **text fields** provided on the page.

New Group

Group path

http://gitlab.reinvestment.com/IT/Group2/

open-source

Group name

Description

Group avatar

Choose File ...

File name...

The maximum file size allowed is 200KB.

Visibility Level

☒ Private

The group and its projects can only be viewed by members.

☐ Internal

The group and any internal projects can be viewed by any logged in user.

☐ Public

The group and any public projects can be viewed without any authentication.

Some visibility level settings have been restricted by the administrator.

- A group is a collection of several projects
- Members of a group may only view projects they have permission to access
- Group project URLs are prefixed with the group namespace
- Existing projects may be moved into a group

Create group

Cancel

Legend:

Required

Optional

6) Click on the **Create group** button when complete.

Note: You must be an owner of a parent group in order to create subgroups and manage users for that parent group.



REINVESTMENT
FUND

Note: Any users from the parent group will be automatically added to all and new subgroups within that parent group, inheriting the current permission levels. Only the Owner and Master of that group can manage users.

Note: Adding and removing users from subgroups will not affect other subgroups or the parent group. However, changing users in the parent group will affect across the parent and subgroups.

New Group

Group path

http://gitlab.reinvestment.com/IT/Group2/

open-source

Group name

Description

Group avatar

Choose File ...

File name...

The maximum file size allowed is 200KB.

Visibility Level ?

☒ Private

The group and its projects can only be viewed by members.

☐ Internal

The group and any internal projects can be viewed by any logged in user.

☐ Public

The group and any public projects can be viewed without any authentication.

Some visibility level settings have been restricted by the administrator.

- A group is a collection of several projects
- Members of a group may only view projects they have permission to access
- Group project URLs are prefixed with the group namespace
- Existing projects may be moved into a group

Create group

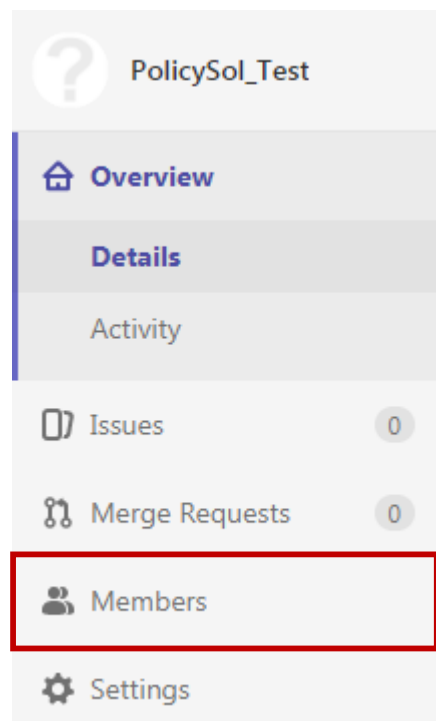
Cancel

Add/Remove Users from Groups and Subgroups

Users with permission levels of Owner or Master can manage users in groups and subgroups. Owners have full administration access in the group compared to Master with only partial administration access. However, users can be added and removed from the parent group and subgroups. Removing users from the subgroup will not affect other subgroups and parent group. On the other hand, removing that user from the parent group will be applied across all the subgroups that the parent group contains.

To add users in the group:

- 1) Click on the **group name**.
- 2) At the left-side bar, click on **Members** tab.



- 3) Under the **Add new member to ...** section, type in the user's name in the text field provided.

Members

Add new member to **IT**

Search for members by name, username, or email, or invite new ones using their email address.

▼

[Read more](#) about role permissions

On this date, the member(s) will automatically lose access to this group and all of its projects.

4) Select the user's **permission level** within this group.

Members

Add new member to **IT**

Guest

Add to group

Search for members by name, username, or email, or invite new ones using their email address.

[Read more](#) about role permissions

On this date, the member(s) will automatically lose access to this group and all of its projects.

5) **(Optional)** Enter the date that the user will no longer have access to this group after the date.

Members

Add new member to **IT**

Guest

Add to group

Search for members by name, username, or email, or invite new ones using their email address.

[Read more](#) about role permissions

On this date, the member(s) will automatically lose access to this group and all of its projects.

6) Click on the **Add to group** button.

Members

Add new member to **IT**

Guest

Add to group

Search for members by name, username, or email, or invite new ones using their email address.

[Read more](#) about role permissions

On this date, the member(s) will automatically lose access to this group and all of its projects.

7) You should see the newly added user's name under the **Existing members** section below.

Existing members

Members with access to **IT** 1



Alysa Truong @truonga

It's you

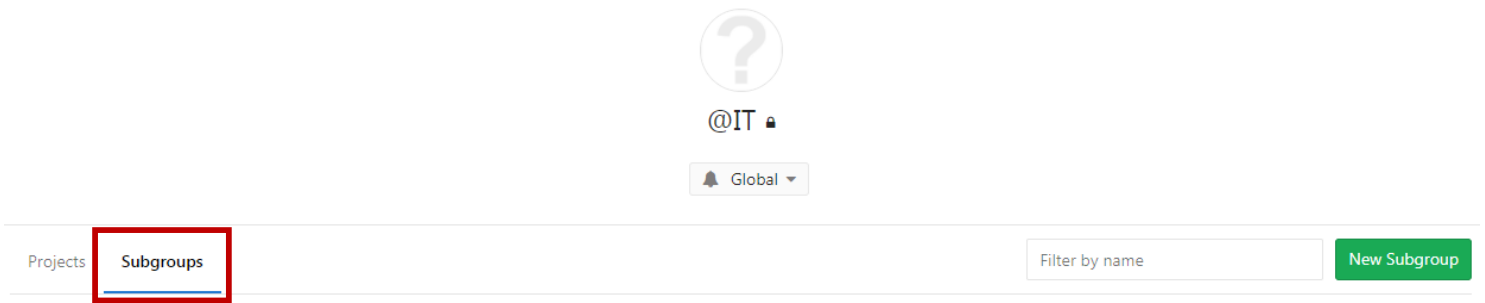
Joined 2 weeks ago

Owner

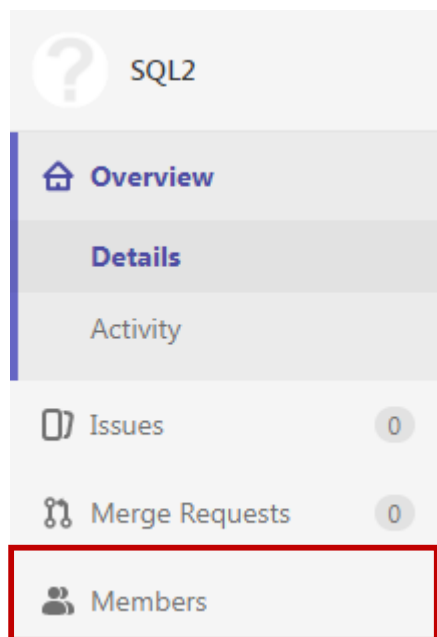
To add users in the subgroup:

1) Click on the **group name**.

- 2) Right next to the Projects tab, click on the **Subgroups** tab.



- 3) Click on the **subgroup name**.
4) On the left side bar, click on the **Members** tab.



- 5) Under the **Add new member to ...** section, type in the user's name.

Members

Add new member to **Group2**

| | | | |
|---|--|---|---|
| <input type="text" value="Search for a user"/> | <input type="text" value="Guest"/> | <input type="text" value="Expiration date"/> | <input type="button" value="Add to group"/> |
| Search for members by name, username, or email, or invite new ones using their email address. | Read more about role permissions | On this date, the member(s) will automatically lose access to this group and all of its projects. | |

6) Select the user's **permission level** within this group.

Members

Add new member to **Group2**

Search for members by name, username, or email, or invite new ones using their email address.

Guest ▼

[Read more](#) about role permissions

Expiration date

On this date, the member(s) will automatically lose access to this group and all of its projects.

Add to group

7) **(Optional)** Enter the date that the user will no longer have access to this group after the date.

Members

Add new member to **Group2**

Search for members by name, username, or email, or invite new ones using their email address.

Guest ▼

[Read more](#) about role permissions

Expiration date

On this date, the member(s) will automatically lose access to this group and all of its projects.

Add to group

8) Click on the **Add to group** button.

Members

Add new member to **Group2**

Search for members by name, username, or email, or invite new ones using their email address.

Guest ▼

[Read more](#) about role permissions

Expiration date

On this date, the member(s) will automatically lose access to this group and all of its projects.

Add to group

9) You should see the newly added user's name under the **Existing members** section below.

Note: Adding users in the subgroup will only give the user access to that particular subgroup; this addition will not be able to access other subgroups within that parent group unless they are added.

Existing members

Find existing members by name



Name, ascending ▼

Members with access to **Group2** 1



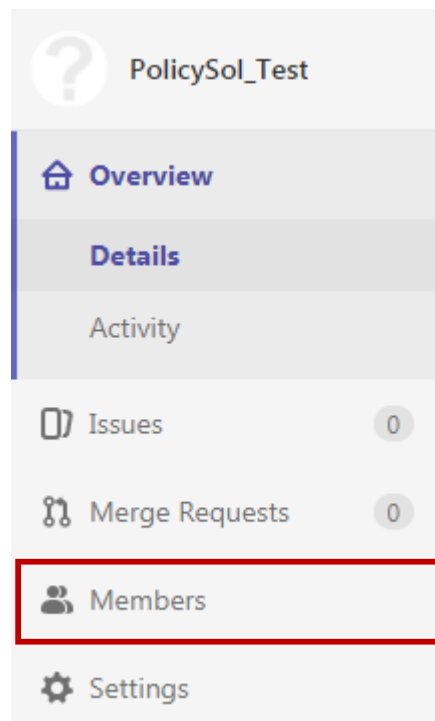
Alysia Truong @truonga It's you

Joined 2 weeks ago

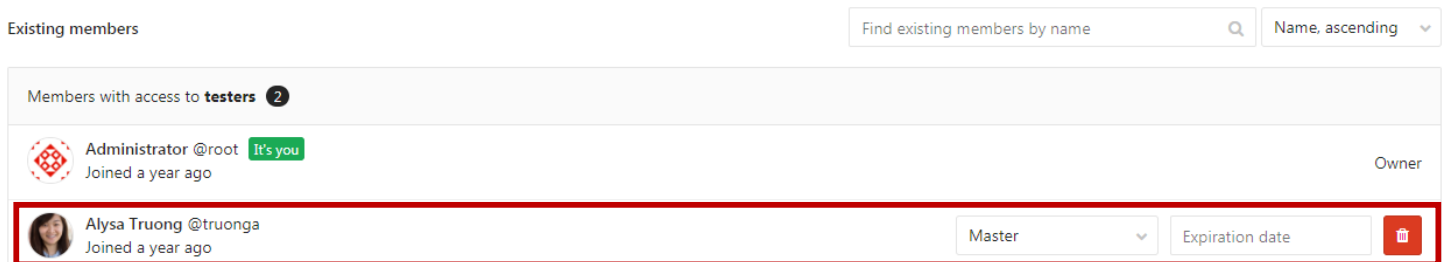
Owner

To remove users in the group:

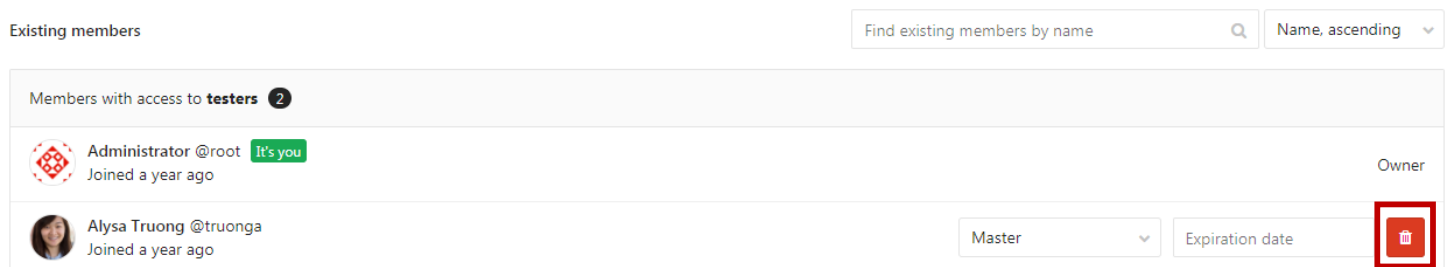
- 1) Click on the **group name**.
- 2) Click on the **Members** tab at the left side bar.



- 3) Under the Existing members section, **locate** the user that you want to remove.



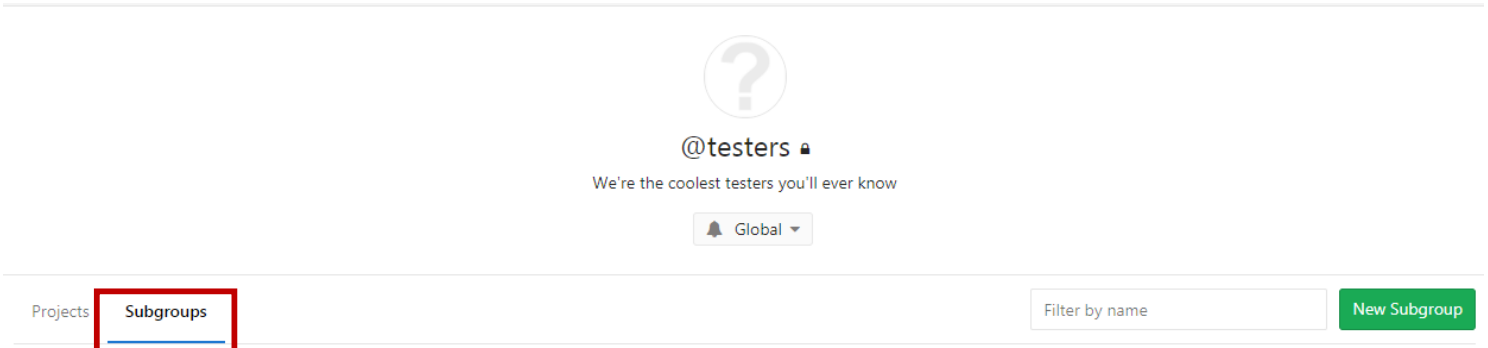
- 4) On the left side of the users list, click on the **trash icon** in the same row as the user's name.



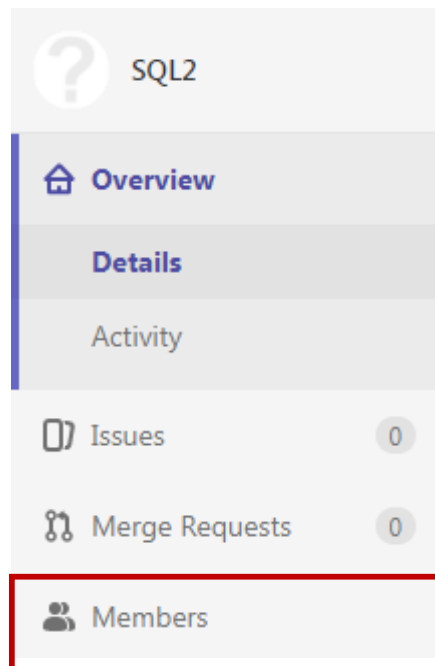
- 5) GitLab will immediately remove the user's access from the group and all of the subgroups that are under the parent group.

To remove users in the subgroup:

- 1) Click on the **group name**.
- 2) On the group home page right next to the Projects tab, click on the **Subgroups** tab.



- 3) Click on the **subgroup name**.
- 4) Click on the **Members** tab on the left side bar.







5) Under the Existing members section, **locate** the user that you want to remove.

Existing members

Find existing members by name Name, ascending

Members with access to **SQL2** **3**





| | | | |
|--|---|----------------------------|--|
|  | Administrator @root It's you Joined 2 weeks ago | | Owner |
|  | Alysa Truong @truonga · testers Joined a year ago | | Master |
|  | VM-test @VMtest Joined less than a minute ago | Guest <input type="text"/> | Expiration date <input type="text"/>  |

6) On the left side of the users list, click on the **trash icon** in the same row as the user's name.

Existing members

Find existing members by name Name, ascending

Members with access to **SQL2** **3**

| | | | |
|--|---|----------------------------|--|
|  | Administrator @root It's you Joined 2 weeks ago | | Owner |
|  | Alysa Truong @truonga · testers Joined a year ago | | Master |
|  | VM-test @VMtest Joined less than a minute ago | Guest <input type="text"/> | Expiration date <input type="text"/>  |

7) GitLab will immediately remove the user's access from the subgroup.

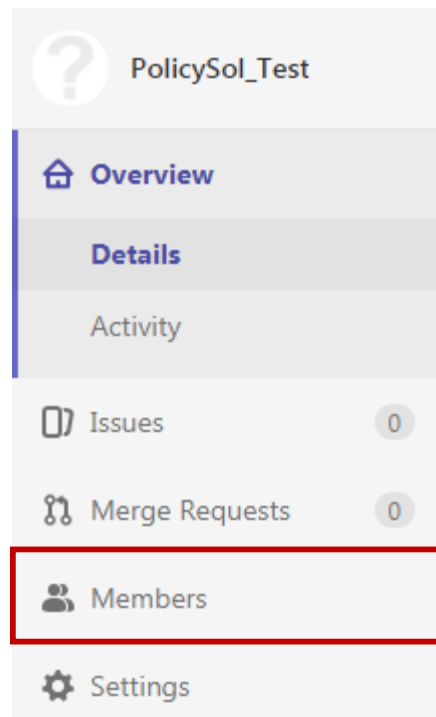
Note: Removing user from the subgroup will only block that user from accessing that current subgroup. The user will continue to have access to the parent group and/or other subgroups unless they are removed either from the parent group or individual subgroups.

Change User's Permission Levels in Groups/Subgroups

Users in groups and subgroups may have completely different permission levels due to the organization or required project contents. However, a user can start off with a low permission level in the group and eventually, they would start to build up their work requirements which will then require a higher permission level to do more things with the project. The owner of the parent group is the only user that can change user's permission levels within groups and subgroups.

To change user permission levels in groups:

- 1) Click on the **group name**.
- 2) At the left side bar, click on the **Members** tab.



- 3) Under the **Add new members to ...** section, type in the user's name in the text field provided.

Members

Add new member to **SQL2**

Search for members by name, username, or email, or invite new ones using their email address.

[Read more](#) about role permissions

On this date, the member(s) will automatically lose access to this group and all of its projects.

- 4) Select the **new permission level** from the drop-down list.

Members

Add new member to **SQL2**

Search for members by name, username, or email, or invite new ones using their email address.

Guest ▼
[Read more](#) about role permissions

Expiration date
On this date, the member(s) will automatically lose access to this group and all of its projects.

Add to group

- 5) (Optional) Enter the **date** that this user will no longer have access to the subgroup after the date chosen.

Members

Add new member to **SQL2**

Search for members by name, username, or email, or invite new ones using their email address.

Guest ▼
[Read more](#) about role permissions

Expiration date
On this date, the member(s) will automatically lose access to this group and all of its projects.

Add to group

- 6) Click on the **Add to group** button.

Members

Add new member to **SQL2**

Search for members by name, username, or email, or invite new ones using their email address.

Guest ▼
[Read more](#) about role permissions




Expiration date
On this date, the member(s) will automatically lose access to this group and all of its projects.

Add to group

- 7) This will update the user's permission levels in all subgroups that is under the parent group. You should see the user with their new permission level under the **Existing members** section.

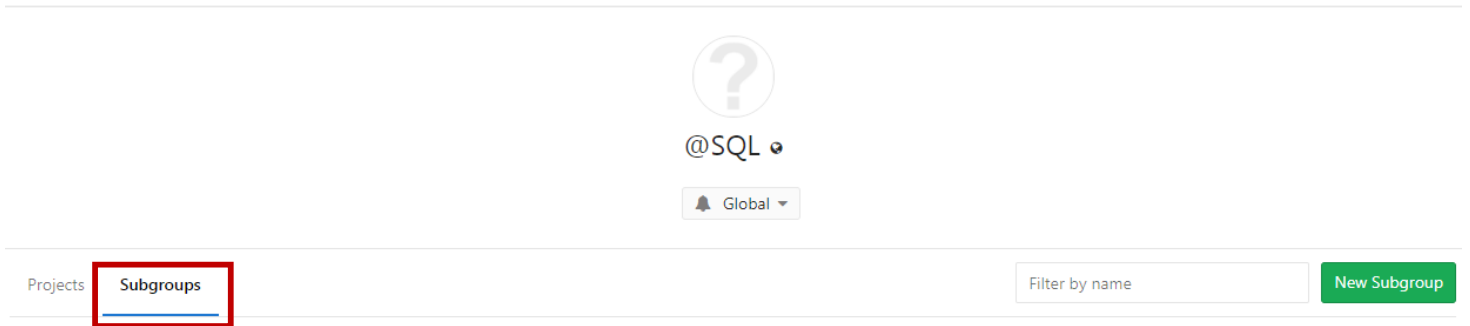
Existing members

Find existing members by name Name, ascending ▼

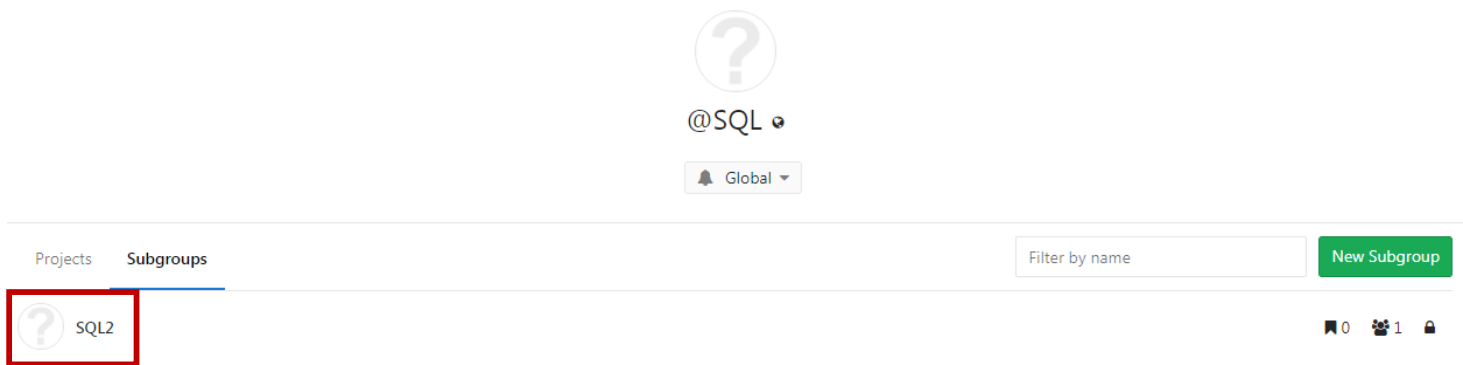
| Members with access to SQL2 3 | | | |
|---|---|---------|---|
|  | Administrator @root It's you Joined 2 weeks ago | | Owner |
|  | Alysa Truong @truonga · testers Joined a year ago | | Master |
|  | VM-test @VMtest Joined less than a minute ago | Guest ▼ | Expiration date <input type="text"/> <input type="button" value="🗑"/> |

To change user permission levels in subgroups:

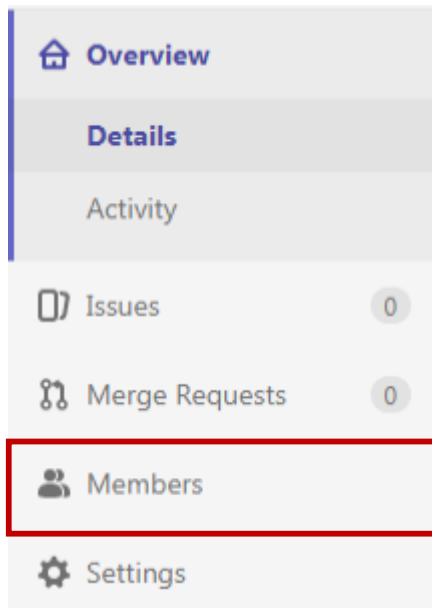
- 1) Click on the **group name**.
- 2) Click on the **Subgroup** tab.



- 3) Click on the **subgroup name** that you want to make changes.



4) At the left-side bar header, click on **Members** tab.



5) Under the **Add new member to ...** section, type in the user's name in the text field provided.

Members

Add new member to **SQL2**

Search for members by name, username, or email, or invite new ones using their email address.

Guest ▼

[Read more](#) about role permissions

Expiration date

On this date, the member(s) will automatically lose access to this group and all of its projects.

Add to group

6) Select the **new permission level** from the drop-down list.

Members

Add new member to **SQL2**

Search for members by name, username, or email, or invite new ones using their email address.

Guest ▼

[Read more](#) about role permissions

Expiration date

On this date, the member(s) will automatically lose access to this group and all of its projects.

Add to group

- 7) (Optional) Enter the **date** that this user will have no longer have access to the subgroup after the date chosen.

Members

Add new member to **SQL2**

Guest ▼

Add to group

Search for members by name, username, or email, or invite new ones using their email address.

[Read more](#) about role permissions

On this date, the member(s) will automatically lose access to this group and all of its projects.

- 8) Click on the **Add to group** button.

Members

Add new member to **SQL2**

Guest ▼

Add to group

Search for members by name, username, or email, or invite new ones using their email address.

[Read more](#) about role permissions

On this date, the member(s) will automatically lose access to this group and all of its projects.




- 9) This will overwrite the old permission level that the user was given in the parent group. You should see the user with their new permission level under the **Existing members** section.

Existing members

Find existing members by name

Name, ascending ▼

Members with access to **SQL2** 3

| | | |
|---|--|-----------|
|  | Administrator @root It's you Joined 2 weeks ago | Owner |
|  | Alysa Truong @truonga · testers Joined a year ago | Master |
|  | VM-test @VMtest · testers / SQL Joined a month ago | Developer |

Note: Only Owner of the subgroup/parent can manage user's permission levels.

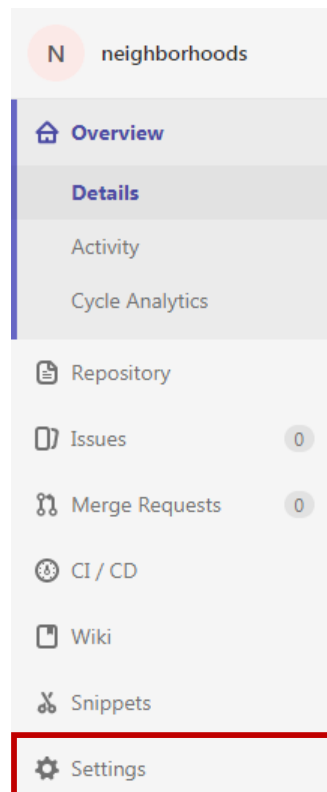
Note: Changing the user's permission levels in the subgroup will not affect the parent group's permission levels unless the owner changes the permission levels in the parent group.

Transfer Projects to Groups/ Subgroups

Sometimes, personal projects that users create would end up extending to multiple people to work on as a collaboration. GitLab provide an easy way to transfer the existing projects over to groups or subgroups without exporting and importing back.

To transfer projects to groups/subgroups:

- 1) Locate the **project repository** in GitLab.
- 2) Click on the **project name**.
- 3) Click on **Settings** option on the left-side header.



- 4) Scroll down the page to find the **Transfer project to new group** section.

Transfer project to new group

Please select the group you want to transfer this project to in the dropdown to the right.

Select a new namespace

- Be careful. Changing the project's namespace can have unintended side effects.
- You can only transfer the project to namespaces you manage.
- You will need to update your local repositories to point to the new location.
- Project visibility level will be changed to match namespace rules when transferring to a group.

Transfer project

5) Click on the **drop-down list** to show all of the groups and subgroups that you are a member of.

Transfer project to new group

Please select the group you want to transfer this project to in the dropdown to the right.

Select a new namespace

- Be careful. Changing the project's namespace can have unintended side effects.
- You can only transfer the project to namespaces you manage.
- You will need to update your local repositories to point to the new location.
- Project visibility level will be changed to match namespace rules when transferring to a group.

Transfer project

6) Select the **group/subgroup** name.

Select a new namespace

Groups

testers

testers / GIS

testers / SQL

testers / SQL / SQL2

Users

Administrator

7) Click on the **Transfer project** button.

Transfer project to new group

Please select the group you want to transfer this project to in the dropdown to the right.

Select a new namespace

- Be careful. Changing the project's namespace can have unintended side effects.
- You can only transfer the project to namespaces you manage.
- You will need to update your local repositories to point to the new location.
- Project visibility level will be changed to match namespace rules when transferring to a group.

Transfer project

- 8) Follow the instruction on the pop-up window by typing the **project name** on the text field provided.

Confirmation required ×

You are going to transfer testers / SQL / neighborhoods to another owner. Are you ABSOLUTELY sure?

This action can lead to data loss. To prevent accidental actions we ask you to confirm your intention.
Please type **neighborhoods** to proceed or close this modal to cancel.

Confirm

- 9) Click on the **Confirm** button.

Note: The options that have multiple forward slashes are considered to be subgroups where the groups that does not have any forward slashes are parent groups.

Confirmation required ×

You are going to transfer testers / SQL / neighborhoods to another owner. Are you ABSOLUTELY sure?

This action can lead to data loss. To prevent accidental actions we ask you to confirm your intention.
Please type **neighborhoods** to proceed or close this modal to cancel.

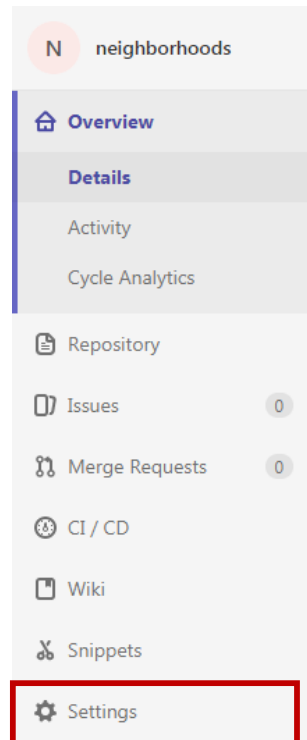
Confirm

Share Projects with Group/Subgroup

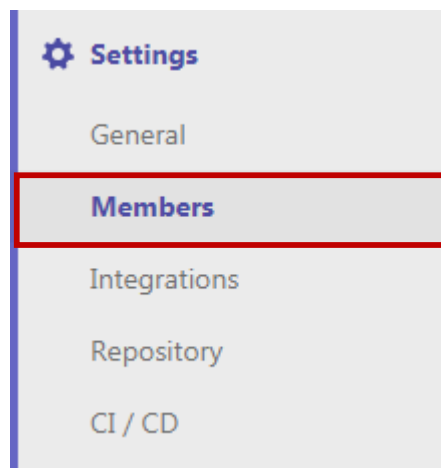
There are times where a group or subgroup may need access to the projects of another groups or subgroups to complete their tasks. GitLab offers the option for owners of the group/subgroup to share their project without exporting and importing online and offline.

To share project with group:

- 1) Click on the **project name**.
- 2) On the left side, click on **Settings** tab.



- 3) At the second-level header under Settings, click on **Members** tab.



4) Click on the **Share with group** tab.

Project members

You can add a new member to **neighborhoods** or share it with another group.

| Add member | Share with group |
|--|------------------|
| Select a group to share with | |
| <input type="text" value="Search for a group"/> | |
| Max access level | |
| <input type="text" value="Developer"/> | |
| Read more about role permissions | |
| Access expiration date | |
| <input type="text" value="Expiration date"/> | |
| <input type="button" value="Share"/> | |

5) Click on the **first drop-down list** to select the group that you want to share the project with.

Project members

You can add a new member to **neighborhoods** or share it with another group.

| Add member | Share with group |
|--|------------------|
| Select a group to share with | |
| <input type="text" value="Search for a group"/> | |
| Max access level | |
| <input type="text" value="Developer"/> | |
| Read more about role permissions | |
| Access expiration date | |
| <input type="text" value="Expiration date"/> | |
| <input type="button" value="Share"/> | |

6) Select the **permission levels** that the group will have to access the project.

Project members

You can add a new member to **neighborhoods** or share it with another group.

| Add member | Share with group |
|--|------------------|
| Select a group to share with | |
| <input type="text" value="Search for a group"/> | |
| Max access level | |
| <input type="text" value="Developer"/> | |
| Read more about role permissions | |
| Access expiration date | |
| <input type="text" value="Expiration date"/> | |
| <input type="button" value="Share"/> | |

7) (Optional) Enter the **date** that the group will no longer have access to the project after the date.

Project members

You can add a new member to **neighborhoods** or share it with another group.

| Add member | Share with group |
|--|------------------|
| Select a group to share with | |
| <input type="text" value="Search for a group"/> | |
| Max access level | |
| <input type="text" value="Developer"/> | |
| Read more about role permissions | |
| Access expiration date | |
| <input type="text" value="Expiration date"/> | |
| <input type="button" value="Share"/> | |

8) Click on the **Share** button.

Project members

You can add a new member to **neighborhoods** or share it with another group.

| Add member | Share with group |
|--|------------------|
| Select a group to share with | |
| <input type="text" value="Search for a group"/> | |
| Max access level | |
| <input type="text" value="Developer"/> | |
| Read more about role permissions | |
| Access expiration date | |
| <input type="text" value="Expiration date"/> | |
| <input type="button" value="Share"/> | |

9) You should see the group name under the **Existing members and groups** section below.

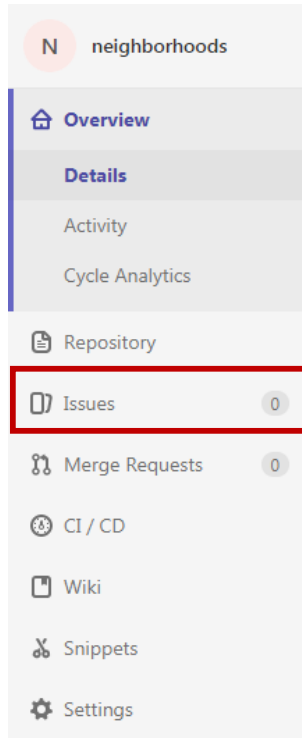
Note: The group list will only show the list of groups that the user have access or permission.

Create Issues

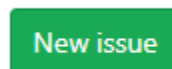
Creating issues will allow users to know the type of errors or problems that are consistently showing up in the project. This will create an area where users can track and manage the list of to-dos that may have certain level of issues.

To create issues:

- 1) Click on the **project name**.
- 2) At the left-side bar, click on **Issues** tab.



- 3) Click on the **New issue** button.



- 4) Follow the instructions below to submit the issue:
 - a. Type in the **title** of the issue.

New Issue

Title

Add [description templates](#) to help your contributors communicate effectively!

b. **Describe** the issue in the text field provided.

Description

Write Preview

Write a comment or drag your files here...

Markdown and slash commands are supported

Attach a file

☐ This issue is confidential and should only be visible to team members with at least Reporter access.

c. Select a **user** to be assigned to the issue.

Assignee Unassigned Assign to me Due date Select due date

Milestone Milestone

Labels Labels

Submit issue

Cancel

d. (Optional) Enter the **date** that this issue must be completed by.

Assignee Unassigned Assign to me Due date Select due date

Milestone Milestone

Labels Labels

Submit issue

Cancel

e. Select the **labels** that briefly describe the issue.

Assignee Unassigned Assign to me Due date Select due date

Milestone Milestone

Labels Labels

Submit issue

Cancel

5) Click on the **Submit issue** button.

| | | | | |
|-----------|---|------------------------------|----------|--|
| Assignee | <input type="text" value="Unassigned"/> | Assign to me | Due date | <input type="text" value="Select due date"/> |
| Milestone | <input type="text" value="Milestone"/> | | | |
| Labels | <input type="text" value="Labels"/> | | | |

Submit issue

Cancel

Create Issue Labels

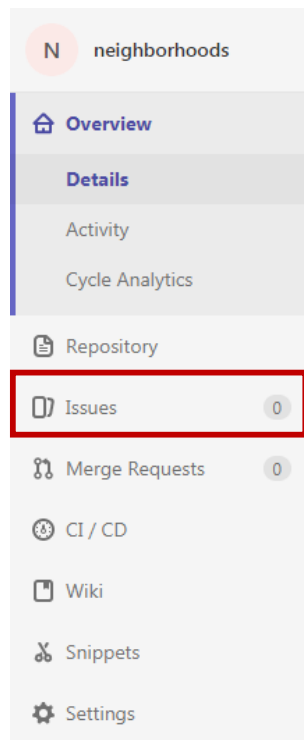
Creating labels for issues helps keep people organized and easily track the types of issues as well as its status of resolution. Labels allows users to quickly glance at the issue on the broader scale to gain a good amount of information of what the issue is about. This way, those issues can be directed to someone who has more knowledge about certain things and can quickly access that issue.

Note: Only Administrator can create global labels. However, the global labels that are newly added will not automatically appeared if the projects existed before creating the labels. They will need to manually add it in the project repository.

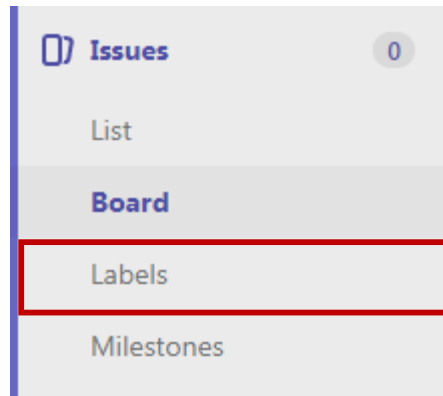
Note: Creating issue labels in the steps below only creates the labels for the issues within that certain project. This means that these labels are not global and must be created in every project.

To create issue labels:

- 1) Click on the **project name**.
- 2) At the left-side bar header, click on the **Issues** option.



3) Click on the **Labels** option.



4) On the right side, click on the **New label** button.



5) Type in the **name** of the label.

New Label

A screenshot of the 'New Label' form. The form has a 'Title' field, a 'Description' field, and a 'Background color' section. The 'Title' field is highlighted with a red rectangular border. Below the 'Background color' section, there is a row of 16 color swatches. At the bottom of the form, there are 'Create label' and 'Cancel' buttons.

6) (Optional) Enter the **description** of the label.

New Label

A screenshot of the 'New Label' form, similar to the one above. In this view, the 'Description' field is highlighted with a red rectangular border. The 'Title' field is also visible and contains a cursor. The 'Background color' section and the row of color swatches are also visible. At the bottom, there are 'Create label' and 'Cancel' buttons.

7) Select a **color** to represent the new label visually.

New Label

Title

Description

Background color

#428BCA

Choose any color.
Or you can choose one of suggested colors below

Create label

Cancel

8) Click on the **Create label** button.

New Label

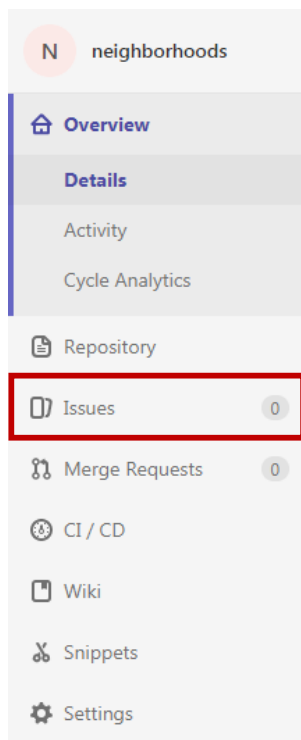
[illegible]

Using Issues Board/List

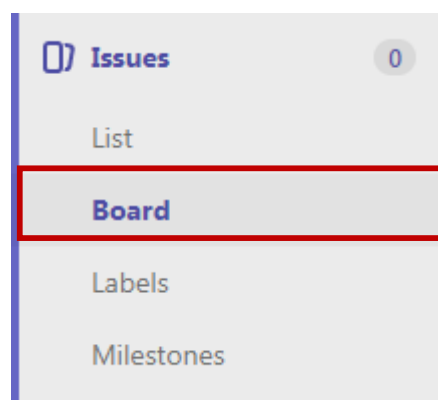
Using the issue board/list will help keep users organized and visualize the current progress of the project in terms of things to do on the list or fixing a certain problem. Users can view and decide between themselves or with group about certain items with statuses, such as *not started*, *in progress*, or *completed*.

To add a new category in the issue board:

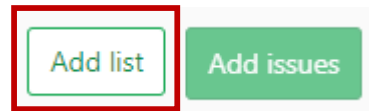
- 1) Click on the **project name**.
- 2) Click on the **Issues** option on the left-side menu bar.



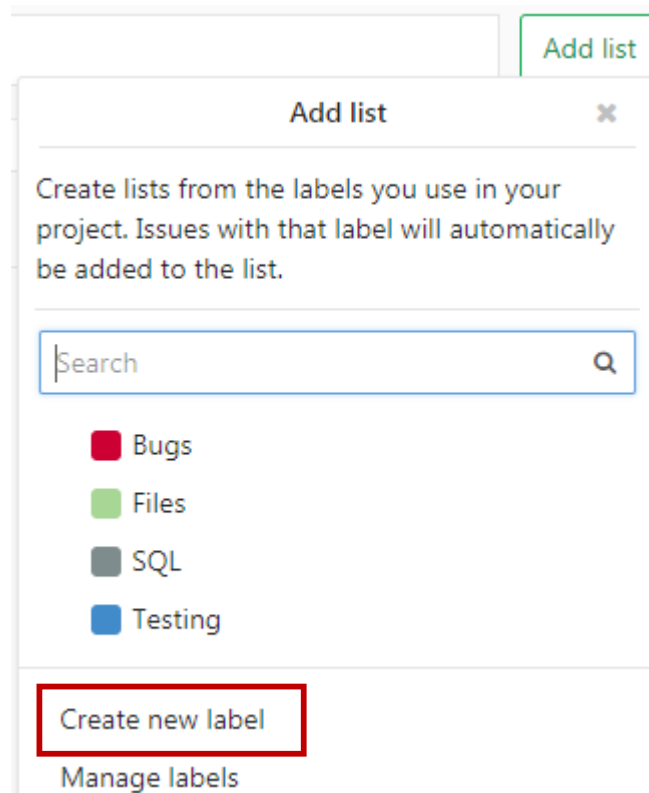
- 3) Click on the **Board** tab underneath the first header.



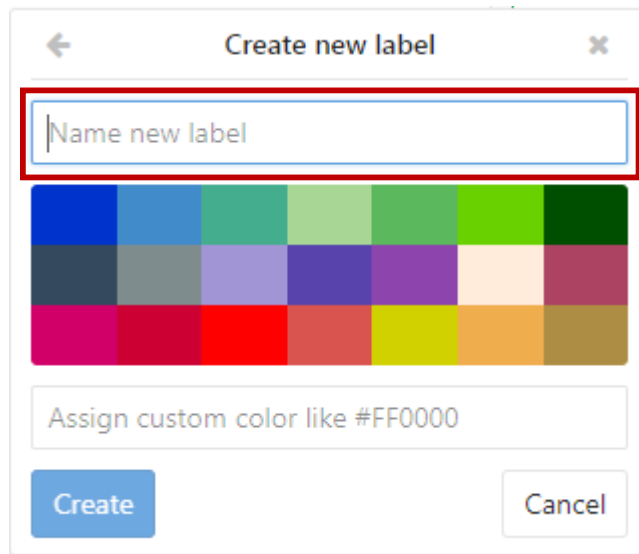
- 4) On the right side of the page, click on the **Add list** button.



- 5) Select **create new label** option.



6) Enter the new label **name** into the text field provided.

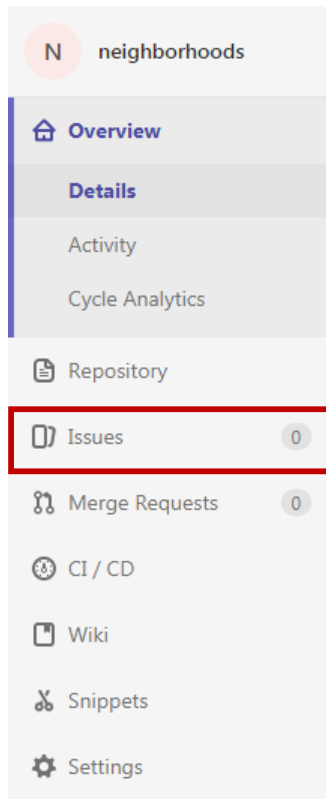


7) Choose the **color** to represent the label.

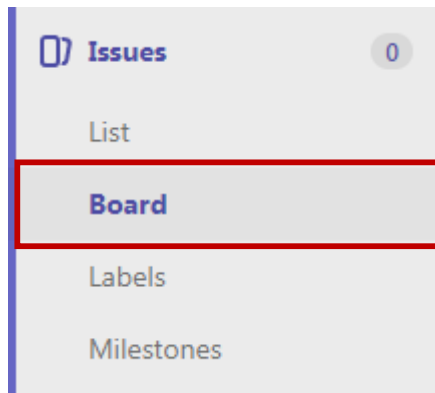
8) Click on the **Create** button.

To move the category around in the issue board:

- 1) Click on the **project name**.
- 2) Click on the **Issues** option.



3) Click on the **Board** tab.



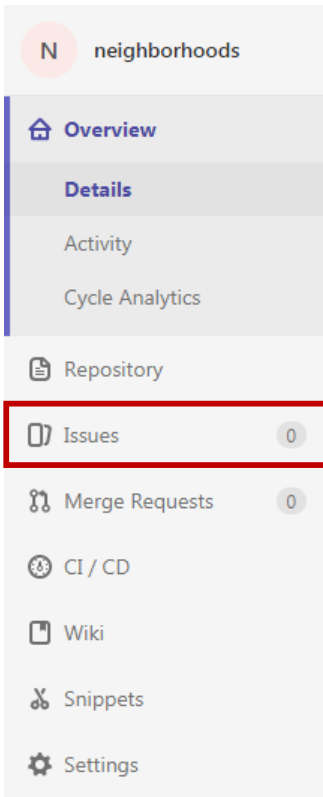
4) Click and hold on the **category** on the board.

5) Drag and drop on the **location** to where you want the category to be placed on the board.

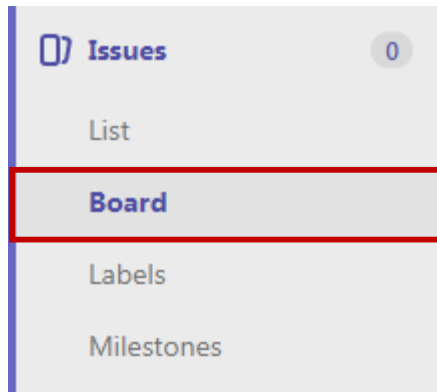
To remove the category in issue board:

1) Click on the **project name**.

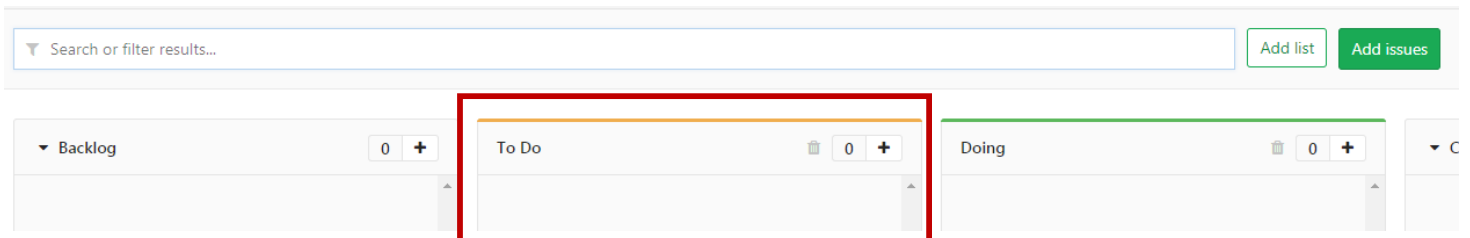
2) Click on the **Issues** option.




3) Click on the **Board** tab.



4) Locate the **category** that you want to remove.



5) Click on the **tiny trash icon** on the same row as the category name.

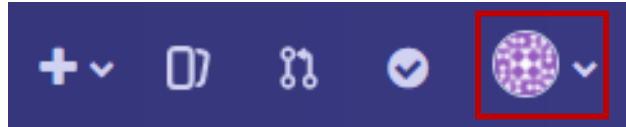
| | | | |
|-------------|--|---|---|
| To Do |  | 0 | + |
| <div></div> | | | |

User Settings

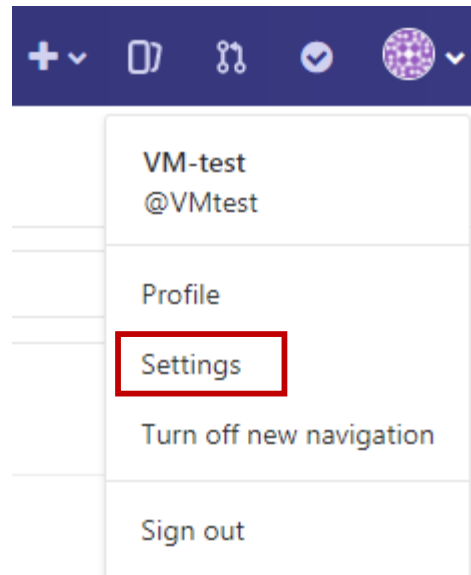
Each users have their own settings where they can configure a bit on their home page appearance as well as the things that they want to view first when logged in. For Administrator's settings, you'll find more information provided in the **Administrator Settings** section in the *Other Mentions*.

To change profile picture:

- 1) Click on the **image** at the top right of the web page.



- 2) Click on the **Settings** tab on the drop-down list.



- 3) In the *Public Avatar* section, click on the **Browse file...** button.

Public Avatar

You can upload an avatar here or change it at gravatar.com



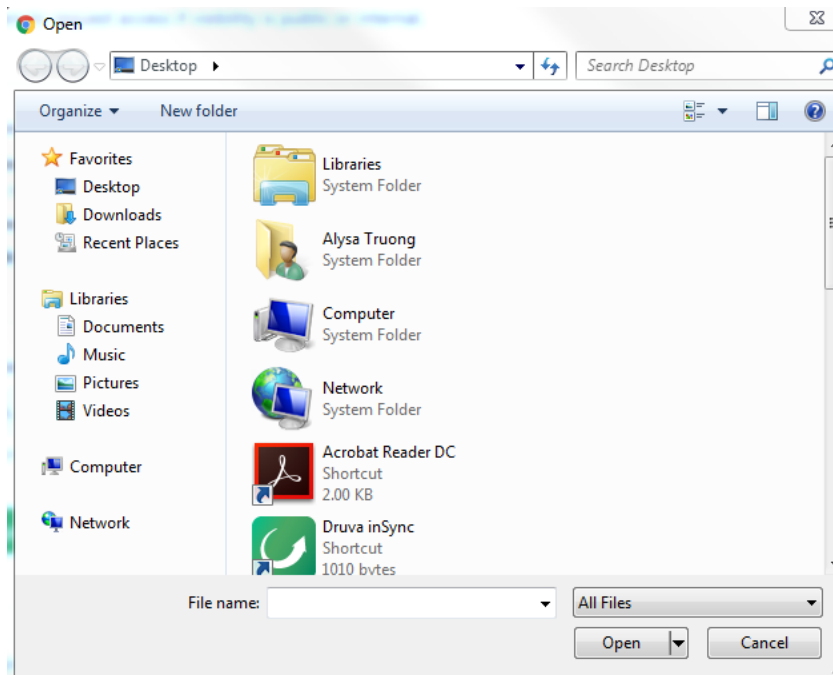
Upload new avatar

Browse file...

No file chosen

The maximum file size allowed is 200KB.

- 4) Locate and choose your **image** from your computer.



- 5) Scroll down to the bottom of the page and click on **Update profile settings** button.

Update profile settings

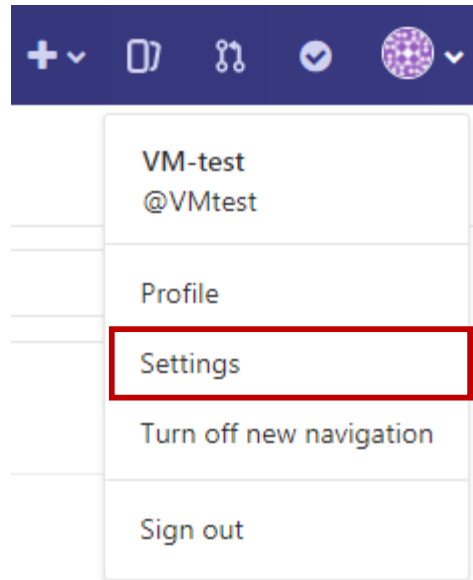
Cancel

To change GitLab appearance:

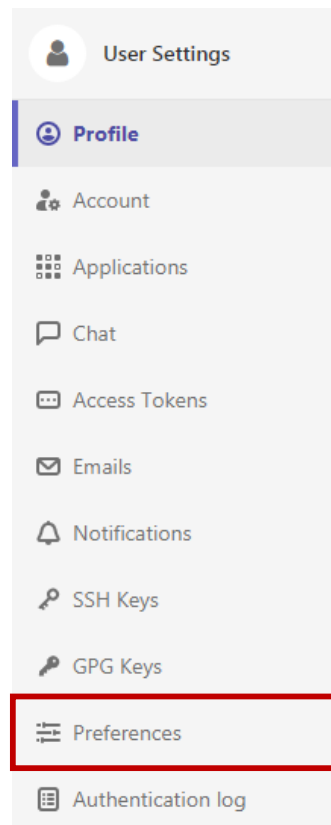
- 1) Click on your **profile picture** at the top right of the page.



- 2) Click on the **Settings** tab.



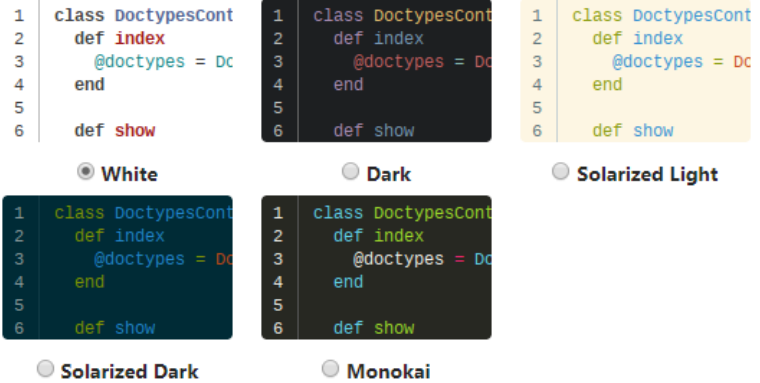
3) At the left-side bar, click on the **Preferences** tab.



4) In the **Syntax highlighting theme** section, select which option you prefer in viewing codes on the GitLab.

Syntax highlighting theme

This setting allows you to customize the appearance of the syntax. [Learn more.](#)



- 5) In the **Behavior** section, select how you would like to view your home page when you first logged in onto GitLab.

Behavior

This setting allows you to customize the behavior of the system layout and default views. [Learn more.](#)

Layout width

Fixed

Choose between fixed (max. 1200px) and fluid (100%) application layout.

Default dashboard

Your Projects (default)

Project home page content

Files and Readme (default)

Choose what content you want to see on a project's home page

- 6) Click on the **Save changes** button when completed.

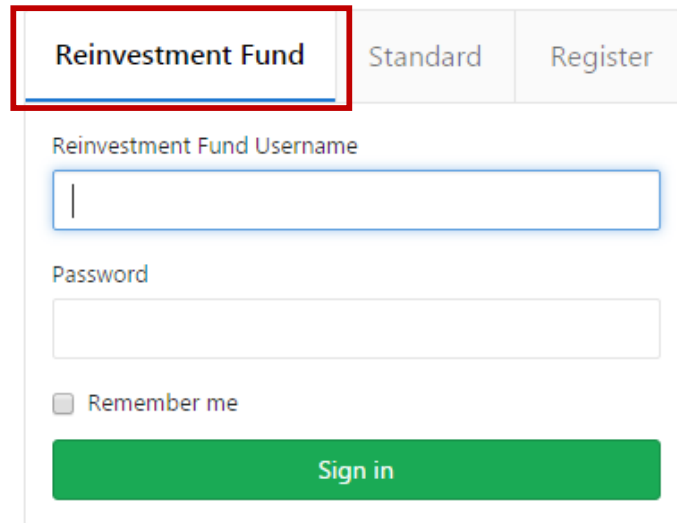
Save changes

Logging into GitLab

This section explains how to login into the GitLab server for the first time as Reinvestment Fund users. When logging into the GitLab server, users will need to use their window credentials. As the users changed their password every 3 months, the new password is used as it syncs up with LDAP.

To log into GitLab:

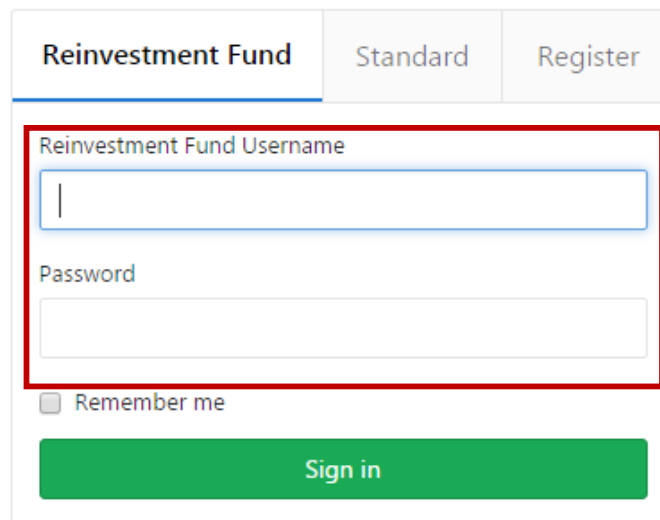
- 1) Head over to [GitLab](#).
- 2) Make sure that the **Reinvestment Fund** tab is selected.



The screenshot shows the GitLab login interface. At the top, there are three tabs: 'Reinvestment Fund', 'Standard', and 'Register'. The 'Reinvestment Fund' tab is highlighted with a red rectangular box. Below the tabs, the form is titled 'Reinvestment Fund Username'. It contains a text input field for the username, a text input field for the password, a checkbox labeled 'Remember me', and a green 'Sign in' button.

Didn't receive a confirmation email? [Request a new one.](#)

- 3) Enter your **window credentials** into the correct text fields provided.



This screenshot is identical to the previous one, showing the GitLab login page with the 'Reinvestment Fund' tab selected. However, a red rectangular box is drawn around the 'Reinvestment Fund Username' text input field and the 'Password' text input field, indicating where the user should enter their window credentials.

Didn't receive a confirmation email? [Request a new one.](#)

- Note:** For first-time users logging into the site, follow the steps below.

Didn't receive a confirmation email? [Request a new one.](#)

-
- REINVESTMENT
FUND

Didn't receive a confirmation email? [Request a new one.](#)

- 7) Once you received a **confirmation email** from the IT Help Desk, head back to **GitLab**. Login back in again with your **window credentials**.
- 8) You should see **GitLab home page** once the page finished loading.

Other Mentions

Below states other sections that it is worth understanding as part of learning GitLab and its workflow.

Permission Levels

Permission levels for users in a group or even in personal projects are very important to make sure that the contents or data contain within are not altered accidentally by someone who should not have access in the first place. As the owner of the project repository and/or group, users can manage users on who has permission to access the group or project as well as what they can do or not do.

Below is the brief summary of the permission levels that users can choose to assign others as:

➤ *Guest*

Users who have guest permission have very limited access to the group or project. This means that guest cannot do much to the project as they only have access to the issues of the project or group. They are able to create new issues, create and view confidential issues, view wiki pages, and leave comments. Therefore, guests have no access to the code at all.

➤ *Reporter*

Users with reporter permission gain the guest ability as well as their own. Unlike guests, reporters can only read the project repository but cannot touch it directly. This means that reporters has read access to the code so they can only fork the project. This option is given for those that you do not want to make changes directly but still want to give access to the code and option to work on their own fork.

➤ *Developer*

Developer permissions can read and write to the project repository directly. This means that developer are not forced to use the merge request when making changes to the project as Reporters are required if they want to make any edits. The best way to remember the permission of developers is that they have regular access to the project in reading and writing; nothing more, nothing less.

➤ *Master*

Masters have the same ability as developers but with partial administrative access. This means that Master can read and write to the project repository (similar to developer permissions) but also approve or reject merge requests on the Owner's behalf. They can edit the project as well as add new members to the group/project.

➤ *Owner*

As an owner, the user has complete control over the project and/or group. This means that this user is runs the whole group by adding new users to the group, removing users from the group,

changing permission levels of users, creating new subgroups (if owner of the parent group as well), managing and overseeing the settings of the group and/or project.

Application Integrations

Users can integrate various applications with GitLab that would help provide a smooth and productive workflow between users and projects. The main applications are Slack and Trello. However, it depends on if the team decides they would like to use Slack and/or Trello to help in keeping organizing and communicating between each other at any time of the day.

To read more information about application integrations, please visit the [GitLab](#) webpage.