To clone a repository via sshfs and be able to adjust the folder name, do this:

After moving into the <folder_name> via cd, do

Assume that the following commit-history is given:

```
                    E -- F -- G -- H                 branch <new_feature>

                 /

A -- B -- C -- D -- D’ -- D’’                        branch <main>
```

When merging <main> and <new_feature>, the commits D and D`` will be either auto-merged or there will be a merge conflict. But of course, they will not be lost!

Now assume that a slightly different commit-history is given:

```
                    E -- F -- G -- H                 branch <new_feature>

                 /

                                                     branch <main>

                 \

                    D’ – D’’                         branch <cool_feature>
```

After merging <main> with <new_feature>, which happens via a fast-forwarding[1], we can delete it, as it is to no use to us anymore. The commit structure then looks like:

```
                                                     branch <main>

                 \

                    D’ -- D’’                        branch <cool_feature>
```

Now we are in the first case again.

To create a local branch, type

[1] After having merged a branch into another one, it it still necessary to push.

To make it a remote branch (which then necessarily appears on gitlab.com), use

To delete a branch only locally, type

If the brain was not merged, use

instead.

To delete a branch remotely, which then also implies that the branch is not visible on gitlab.com anymore, additionally use

To resolve a merge conflict, type

Afterwards, confirm with Enter that you want to use vimdiff as default editing tool. The vimdiff display will now resemble the following structure:

If the file has not already existes in BASE, then we need this view:

It is probably easiest to take the merged view and edit it directly. In the vim editor, an entire line can be deleted by pressing D (no control before!).

If I instead wanted the changes from either LOCAL, BASE or REMOTE, I have to do:

or

or

Of course, the merged view can also be edited directly.

Regardless of the chosen method, type

into vim. Afterwards, do not forget to commit and push. And if you want, do

Locally restore file

git restore does not overwrite HEAD, though. For that, a push would be necessary.

To include untracked files for the stash, do

When applying the stash (to make sure that the files are staged also after applying the stash, use --index option):

To remove a particular item from the stash, do

To find out the index number on the stash, we can simply do

To remove all stashes, do

This command is irreversible, though!

```
List all files:
```