



Gitlab Tool v0.0.2

User guide

Powered by Luxoft

## 1. Goal (Purpose to use)

// add goal

## 2. Platforms

1. Windows 7, 8, 8.1, 10;
2. GTK Linux (RHEL 6 or higher)

**Important:** you have to install JRE v1.8.0u45 or higher to run this application

## 3. Start application

There are two ways to launch the application:

1. Double click on jar file (if JRE already installed, application will be launched automatically)
2. Open the terminal (or console for Windows OS), go to the directory with GitlabTool-full.jar file and run the next line

```
java -jar GitlabTool-full.jar
```

In both cases you will see the login dialog.

## 4. Login in



Image 1 – Login window

Input your credentials, choose the Gitlab© server and press Sign in button. After that the message about login in will be shown. If login in is successful, the Clone group window will be shown. If login in is failed, the message about it will be shown.

After the first successful login in the username and last chosen server will be saved in the user home dir

C:\Users\<user\_name>\.GitlabTool for Windows, //home/.GitlabTool for \*nix

Also, the application logs will be stored there.

On the next application launches the last chosen server and username will be shown on the login window.

### ***Choose the server***

There are two types of Gitlab© servers available by default: `gitlab.lgc.com` and `gitlab.com`

User could choose another Gitlab© server. Select 'Other...' item from drop-down server menu. The Server selection window will be shown.



Image 2 – Server selection window

Input the server link, choose API version (v3 is available for now) and press the OK button. If server link is incorrect, the message about it will be shown. If everything is OK, new item will be added in drop-down server menu.

## **5. Group cloning window**

After successful login in the Clone group window will be shown.

At the top of the window we have the main menu, the toolbar and user's name. On the left side of the window we have the list of cloned groups (empty by default). Group list part could be extended.

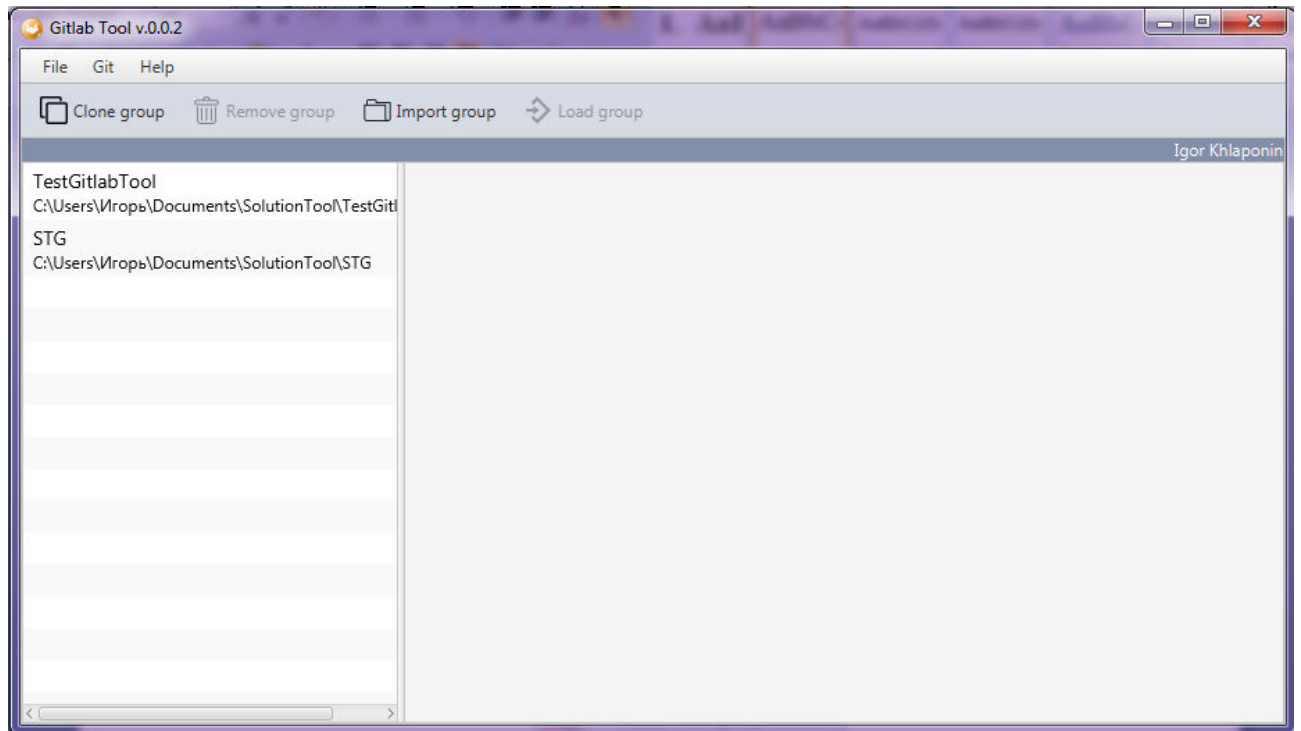


Image 3 –Clone group window

### ***Group cloning***

Press Clone group button on the toolbar (or Git → Clone on the main menu) → the Cloning window will be shown.

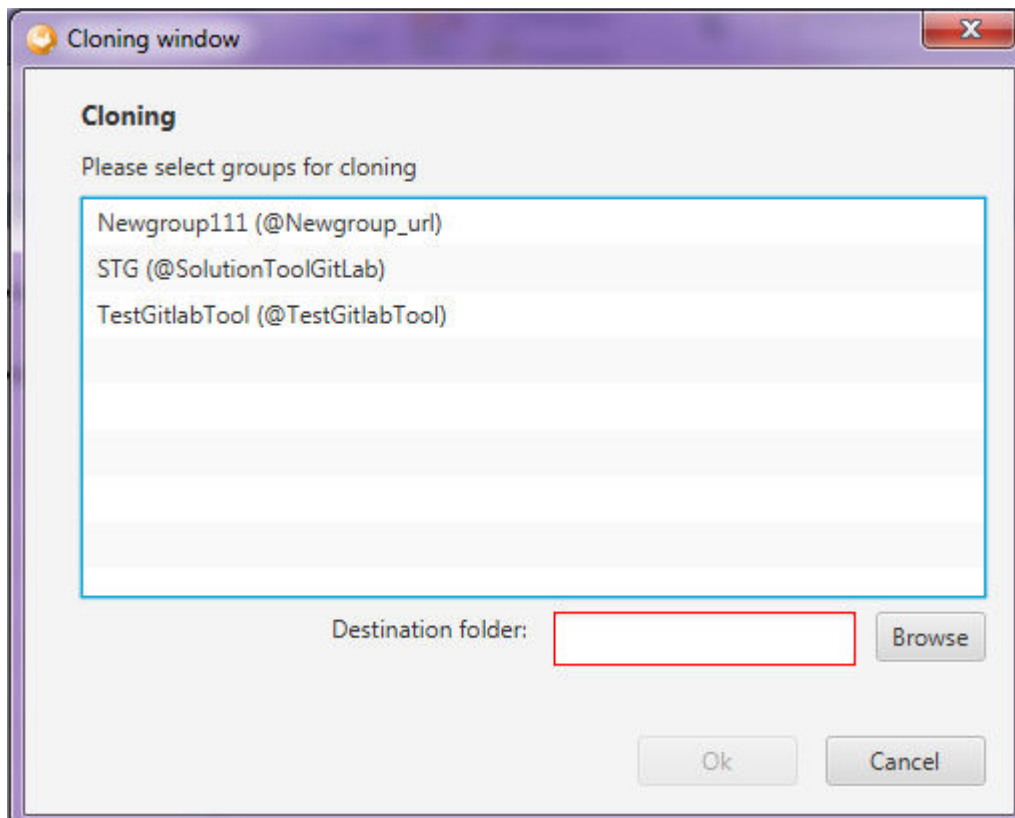


Image 4 – Cloning window

Select the group and destination folder and press OK button. If destination folder wrong, the destination folder field will be highlighted with red color.

After that the clone progress will be shown in separate window.

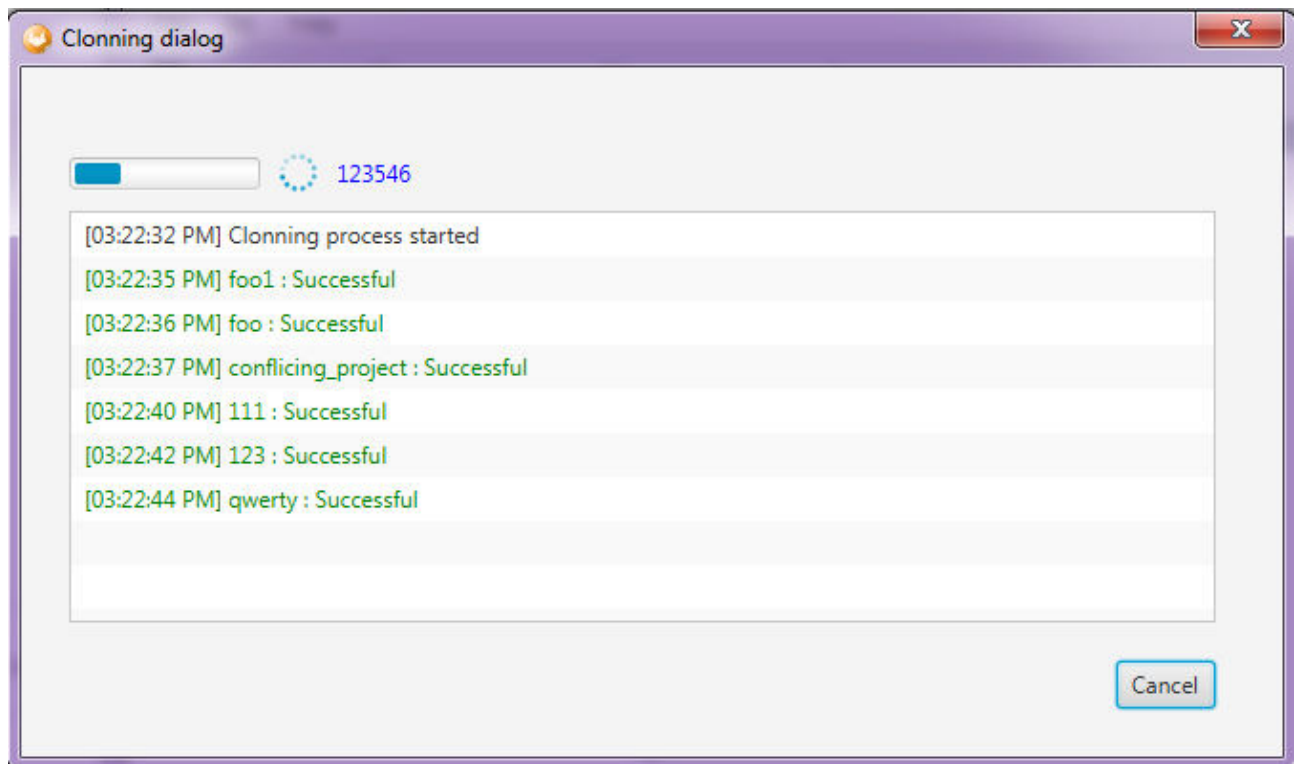


Image 5 – Clone progress dialog

Clone process could be canceled. In this case only already cloned project of the group will be available in the project list.

After finishing the clone process the status dialog with information about cloning will be shown. Press OK button to close Progress dialog. After that, cloned group will be available in the group list.

### ***Import group***

If group already stored on the disk, it could be imported to Gitlab Tool.

Press Import group button on the toolbar and select the folder with existing group on the disk. After that, select Ok button. Group will be added to the group list.

### ***Delete group***

Each group could be deleted from the group list. Select the group and press Remove group button → Remove group window will be shown.

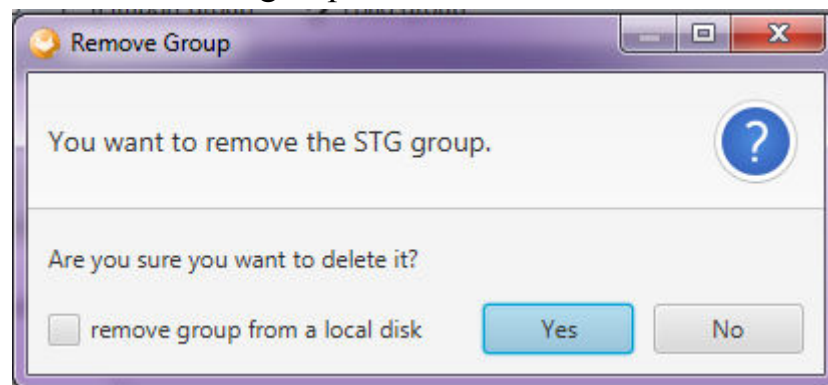


Image 6 – Remove group window

User could delete group only from Gitlab Tool or both: from Gitlab Tool and the disk (toggle 'remove group from a local disk' for it and press 'Yes')

## ***Load group***

Select group and press Load group button or just double click on group item to load the list of group's projects.

After that, main window will be shown.

## **6. Main window**

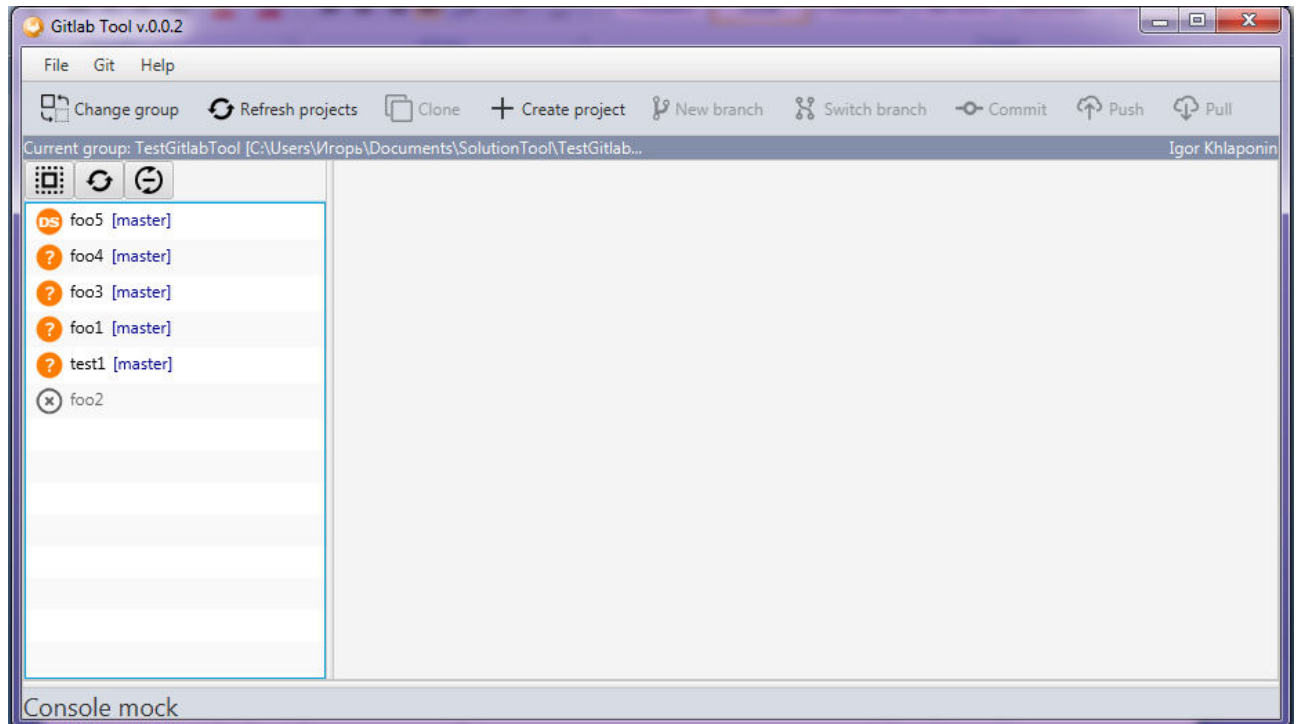


Image 7 – Main window

At the top of the window we have the main menu, the toolbar and the information bar. Information bar contains the name of current group and path for it and, also, the user name.

There is a console part at the bottom of the window (mock for now).

On the left we have projects list with own toolbar.

Projects list toolbar contains three buttons:




- Select all/Deselect all projects



- Refresh projects



- Filter (show/hide) shadow projects.

Shadow projects aren't cloned yet and marked with  sign. They could be cloned later. Gitlab tool does not support any action with shadow projects except cloning.

Each item of the project list consists of three parts: a sign of the project type on the left part, current branch in square brackets on the right part and project name between them.

### ***Change group***

Current group could be changed. Press Change group button → Clone group window will be shown again. After that another group could be chosen.

### ***Clone project***

Shadow project could be cloned by Gitlab Tool.

Select shadow project (or set of shadow projects) and press Clone button on the toolbar (or Git → Clone on the main menu or MB3 → Clone on the project item). After that the clone progress dialog will be shown (Image 5). If clone process successful, cloned projects will be shown on the project list as normal object. There is a set of Git options available now for this object.

### ***Create project***

New project in the group could be created by Gitlab Tool.

Press Create project button on the toolbar → the Create Project Dialog will be shown.

Input the project name and select the project type. After that select Create Project button.

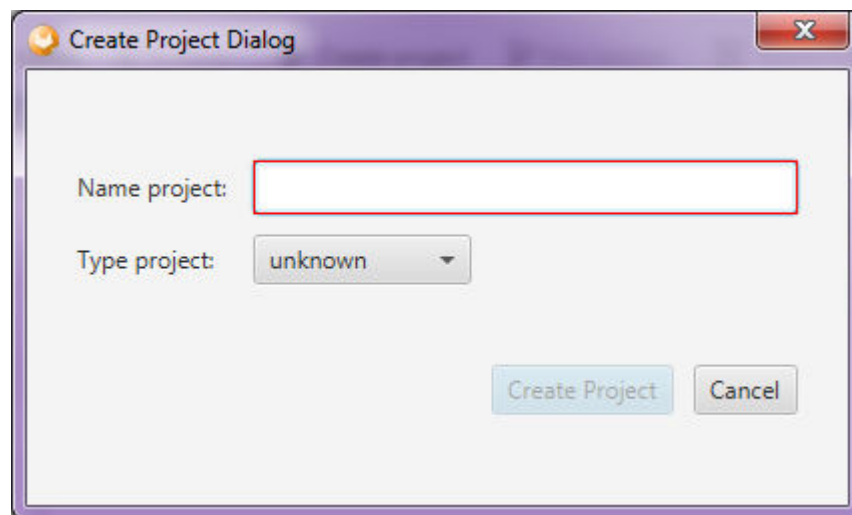


Image 8 – Create project dialog

A new status dialog will be shown after project creation.

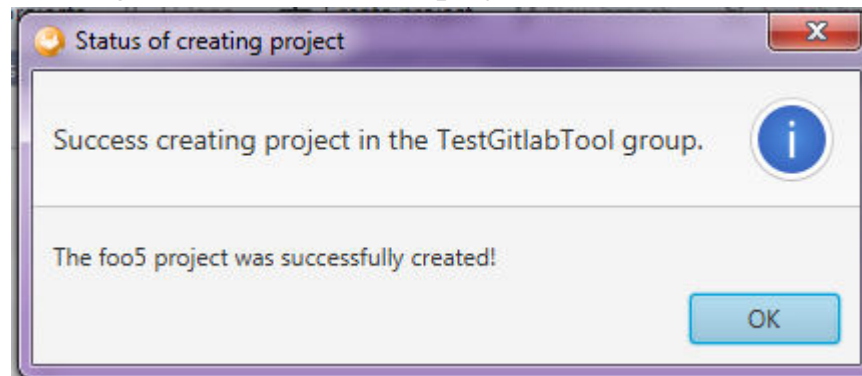


Image 9 – Status dialog

After that, new project will be shown on the projects list.

### ***Create new branch***

Select a project (or set of projects) and press New branch button on the toolbar (or Git → New branch on the main menu) → Create new branch window will be shown.

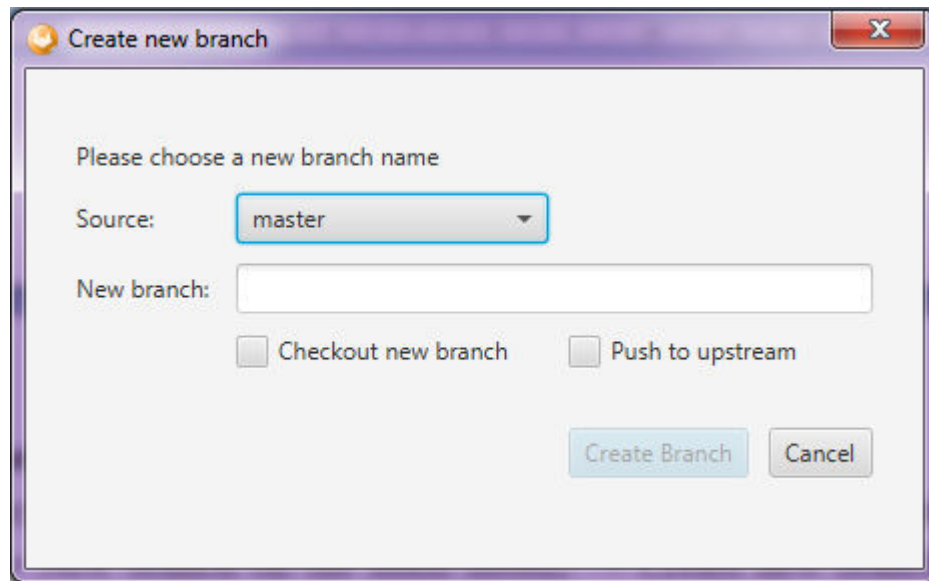


Image 10 – Create new branch window

Select the source branch. If all selected project are on the same branch, this branch will be shown as a source branch. Else if selected project are on the different branches, the source branch field will be empty. In this case user should select the source branch from drop-down menu (it contains only common branches for all selected projects).

Insert the new branch name. If the name isn't correct, the 'New branch' field will be highlighted with red and the 'Create Branch' button will be disabled.

If the name is correct (highlighted with green color) and source branch is chosen, press Create branch button → Status dialog about new branch creation will be shown.

If 'Checkout new branch' checkbox is toggled, the project will be switched on the created branch.

If 'Push to upstream' checkbox is toggled, the project will be switched on the created branch ('Checkout new branch' checkbox should be selected automatically) and new branch will be pushed to the server.

If we switch off the 'Checkout new branch', 'Push to upstream' checkbox should be switched off automatically because we could not push branch before checking out.

After creation of the new branch the status dialog will be shown.

Status dialog should contain the statuses of new branch creation for selected projects. If more than 10 projects were selected, the count of projects with successful branch creation should be shown.



## ***Switch branch***

Select project (or a set of projects) and press Switch branch button on the toolbar (or Git → Switch branch on the main menu) → Switch branch window will be shown.

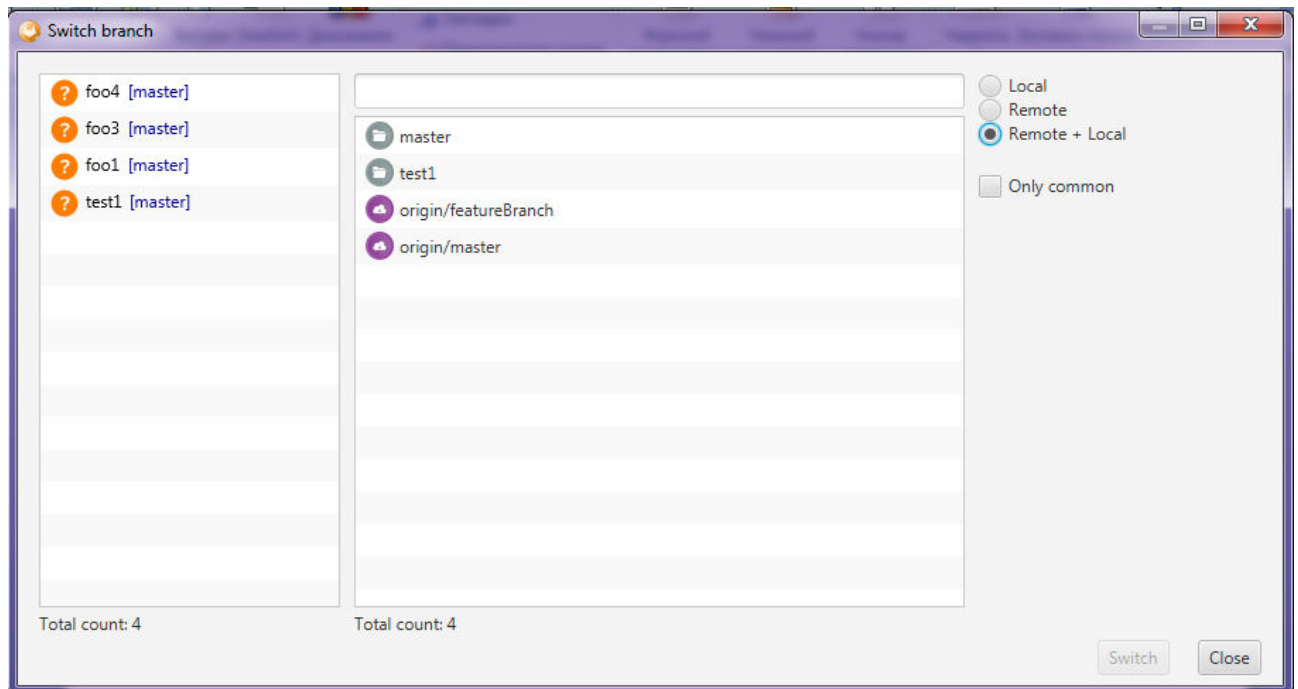


Image 11 – Switch branch window

Switch branch window consists of three parts.

On the left side there is a list of selected projects. A total count of selected projects is shown at the bottom of the list.

On the middle of the window there is a list of branches. List of branches depends on the filter.

On the right side of the window there is a set of filters (radio buttons).

Local button shows only local branches on the branches list.

Remote button shows only remote branches on the branches list.

Remote + Local button shows both the local and the remote branches.

‘Only common’ checkbox filters branches to show only common branches for all selected projects.

Select appropriate branch and press Switch button at the bottom of the window → all selected projects will be switched on the selected branch.

## ***Commit changes***

If some project has uncommitted changes, you could commit them by Gitlab Tool.

Select the project with uncommitted changes and press Commit button on the toolbar (or Git → Commit on the main menu) → Commit and Push dialog will be shown.

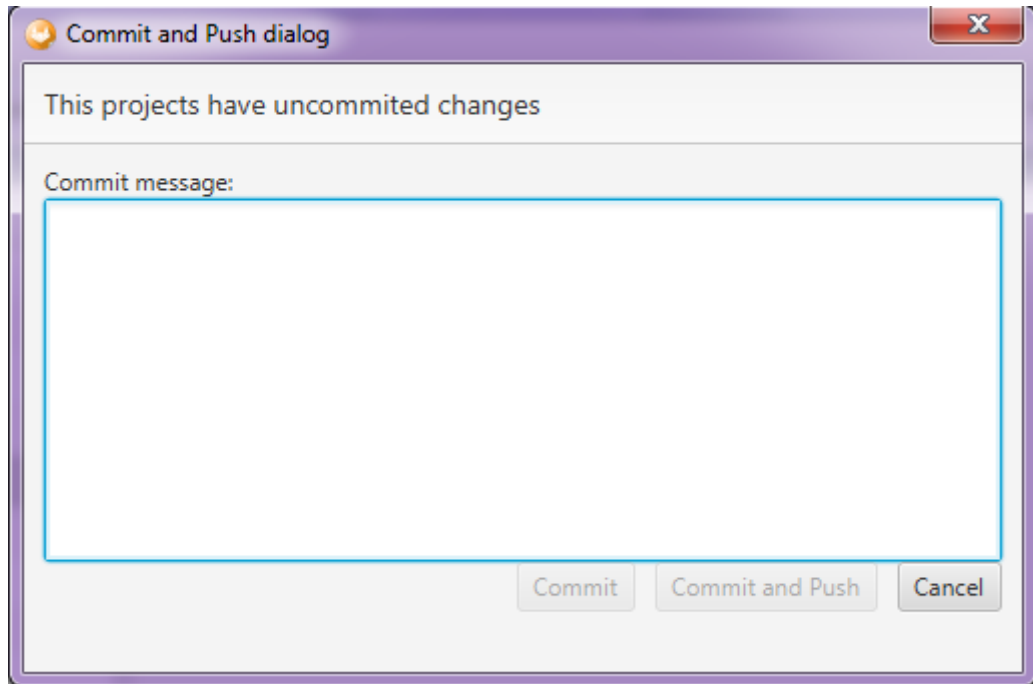


Image 12 – Commit and Push dialog

Input commit message and press Commit or Commit and Push button.

If Commit button pressed, changes will be committed and stored in local repository.

If Commit and Push button pressed, changes will be committed and pushed to server.

After that a new Status dialog will be shown with the result of committing.

#### ***Push to server***

Select the project (or a set of projects) and press Push button on the toolbar (or Git → Push on the main menu). After that, all selected projects will be pushed to upstream and status dialog with results of pushing will be shown.

#### ***Pull from server***

Select the project (or a set of projects) and press Pull button on the toolbar (or Git → Pull on the main menu). After that, the Pull progress dialog will be shown and then Status dialog about pull results.

## 7. Contacts

If you have questions, please contact to Yurii Pitomets

