



UNIVERSITY
OF CRETE

CS-255 – Git Tutorial

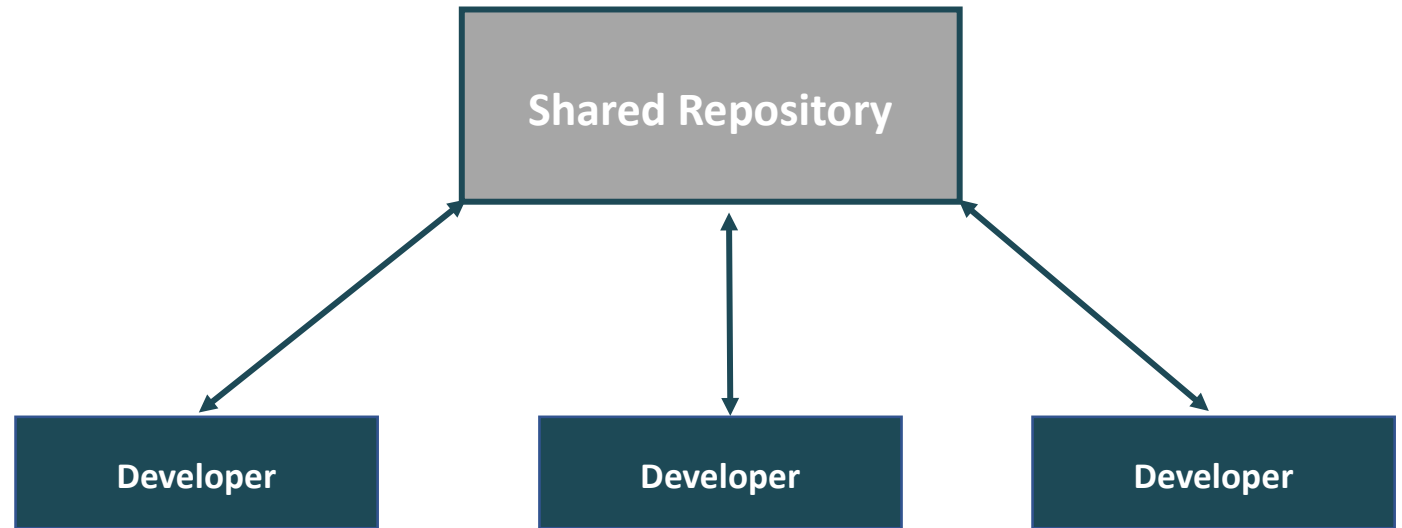
Giorgos Xanthakis
Iacovos Kolokasis
University of Crete
gxanth@csd.uoc.gr
kolokasis@csd.uoc.gr

Outline

- What is git
- How to join CS255 group
- What are ssh keys and how to use them
- How to fork a project
- Make a project private and add TAs as members
- How to use private.py
- How to use git

What is Git?

- Version control system
- Track file history
- Shared among multiple users



How to Login to CSD Git?



- Go to csd giltyab: <https://gitlab-csd.datacenter.uoc.gr>

Computer Science Department, University of Crete



Pilot installation of GitLab, a web-based Git-repository manager.

GitLab Community Edition

Open source software to collaborate on code

Manage Git repositories with fine-grained access controls that keep your code secure. Perform code reviews and enhance collaboration with merge requests. Each project can also have an issue tracker and a wiki.

1

LDAP

Standard

LDAP Username

csdp1039@csd.uoc.gr

Password

.....

☐ Remember me

Sign in

How to Join CS255 Group?

The image illustrates the steps to join the CS255 group on GitLab:

- Step 1:** Navigate to the **Groups** section in the GitLab sidebar. The **Explore groups** option is highlighted, and a red **2** indicates the next step.
- Step 2:** In the **Groups** page, the **CS255 23** group is highlighted in the **Frequently visited** list, and a red **3** indicates the next step.
- Step 3:** On the **CS255_23** group page, the **Request Access** button is highlighted, and a red **4** indicates the final step.

Group Details:

- Group Name:** CS255 23
- Group ID:** 1525
- Description:** CS255 Programming Lab Spring 2023, Department of Computer Science, University of Crete (<https://www.csd.uoc.gr/~hy255/>)
- Instructor:** Prof. Angelos Bilas (<http://www.csd.uoc.gr/~bilas>)

Subgroups and projects:

Project Name	Stars	Last Updated
assign1_2023	0	3 days ago
assign2_2023	0	1 day ago
assign3_2023	0	1 week ago
assign4_2023	0	1 week ago
assign5_2023	0	1 week ago

Generating ssh-keys

- Run: `ssh-keygen -t rsa -b 2048 -C "kolokasis@ics.forth.gr"`
- Press enter in the next three options
- Run: `cat ~/.ssh/id_rsa.pub`
- Copy the printed text

```
[root@desktop ~]# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:vvDbwVF2YHAoSCxnjhNojc4oEenLJCYgGcsGNyI2Gyw root@desktop.example.com
The key's randomart image is:
+---[RSA 2048]---+
|B0o+ o.. .o+
|E=B.+ = . .o .
|*X B . o .
|*+o o . o .
|B . . S .
| o . . .
| . . o
| o o .
| +..
+-----[SHA256]-----+
```

Enter passphrase here, if required

Location of Public Key

Location of Private Key

Add ssh-keys in your Account

The screenshot illustrates the steps to add SSH keys in a GitLab account. It is divided into three numbered sections:

- 1**: The top navigation bar. A red box highlights the user profile icon in the top right corner.
- 2**: The user settings dropdown menu. A red box highlights the 'Preferences' option.
- 3**: The 'User Settings' sidebar. A red box highlights the 'SSH Keys' option.

The main content area shows the 'Preferences' page with a search bar and two sections:

- Navigation theme**: Customizes the appearance of the application header and navigation sidebar. It offers color themes (Indigo, Light Indigo, Blue, Light Blue, Green, Light Green, Red, Light Red) and a 'Dark Mode (alpha)' option.
- Syntax highlighting theme**: Customizes the appearance of the syntax. It shows three preview cards for 'White', 'Dark', and 'Solarized Light' themes, each displaying a code snippet.

Add ssh-keys in your Account

User Settings > SSH Keys

🔍 Search settings

SSH Keys

SSH keys allow you to establish a secure connection between your computer and GitLab.

Add an SSH key

To add an SSH key you need to [generate one](#) or use an [existing key](#).

Key

Paste your public SSH key, which is usually contained in the file '~/.ssh/id_ed25519.pub' or '~/.ssh/id_rsa.pub' and begins with 'ssh-ed25519' or 'ssh-rsa'. Do not paste your private SSH key, as that can compromise your identity.

Typically starts with "ssh-ed25519 ..." or "ssh-rsa ..."

Title

e.g. My MacBook key

Give your individual key a title. This will be publicly visible.

Expires at

mm/dd/yyyy

Key can still be used after expiration.

Add key

4

5

How to Fork a Repository

The screenshot illustrates the process of forking a repository on GitLab. It is divided into three numbered sections:


- 1. Project selection:** The left sidebar shows a list of projects under the group 'CS255_23'. The project 'assign1_2023' is highlighted with a red box and a red '1'.
- 2. Forking the project:** The main content area shows the details of the 'assign1_2023' project. The 'Fork' button is highlighted with a red box and a red '2'.
- 3. Selecting a namespace:** A modal dialog titled 'Select a namespace to fork the project' is shown. It displays a user profile for 'GEORGIOS XANTHAKIS' with a 'Select' button highlighted by a red box and a red '3'.

The interface includes a top navigation bar with 'GitLab', 'Projects', 'Groups', and 'More' menus. The left sidebar contains links for 'Group overview', 'Details', 'Activity', 'Issues', 'Merge requests', 'Packages & Registries', and 'Members'. The main content area shows the project details for 'assign1_2023', including its ID (7665), commit count (1), branch count (1), tag count (0), file size (215 KB), and storage size (215 KB). The 'Fork' button is located in the top right corner of the project details section.

Info About Fork

- You need to fork each assignment **only once!**
- If you want to download your project in different devices (e.g., laptop, desktop PC) you need to clone it

GEORGIOS XANTHAKIS > assign1_2023

 The project was successfully forked.

A

assign1_2023

Project ID: 7790

 ▾

 Star 0

 Fork 0

 1 Commit  1 Branch  0 Tags  133 KB Files  246 KB Storage

Forked from CS255_23 / assign1_2023

master ▾

assign1_2023 /

+ ▾

History

Find file

Web IDE

 ▾

Clone ▾



Initial commit

Jack Kolokasis authored 1 week ago

 README

 Auto DevOps enabled

 Add LICENSE

 Add CHANGELOG

 Add

Clone with SSH

git@gitlab-csd.datacenter.uoc.gr:c

https://gitlab-csd.datacenter.uoc.

Open in your IDE

Visual Studio Code

Name

Last commit

 src

Initial commit

Make your Project Private

- To make your project private
 - Go to Project Settings -> General
 - Visibility
 - Change Project Visibility to Private
-
- To add the TAs in your project
 - Go to Project Settings -> Members
 - Add each TA as a Developer
-
- But we can avoid to do these steps on every assignment

Using private.py

- Run `pip3 install --upgrade python-gitlab --user`
- Copy your access token and the assignment project id

`python3 private.py -t <accessToken> -p <projectID>`

CS100 > assign1

assign1 Project ID: 3278

1 Commit 1 Branch 0 Tags 164 KB Files 164 KB Storage

Hello World

master assign1 / +

History Find file Web IDE Clone

Initial commit
Jack Kolokasis authored 46 minutes ago 6647411b

README Auto DevOps enabled Add LICENSE Add CHANGELOG Add CONTRIBUTING

Name	Last commit	Last update
Makefile	Initial commit	46 minutes ago
README.md	Initial commit	46 minutes ago
private.py	Initial commit	46 minutes ago

README.md

Assignment1: Hello World

Make Repository Private and Add TAs as Members

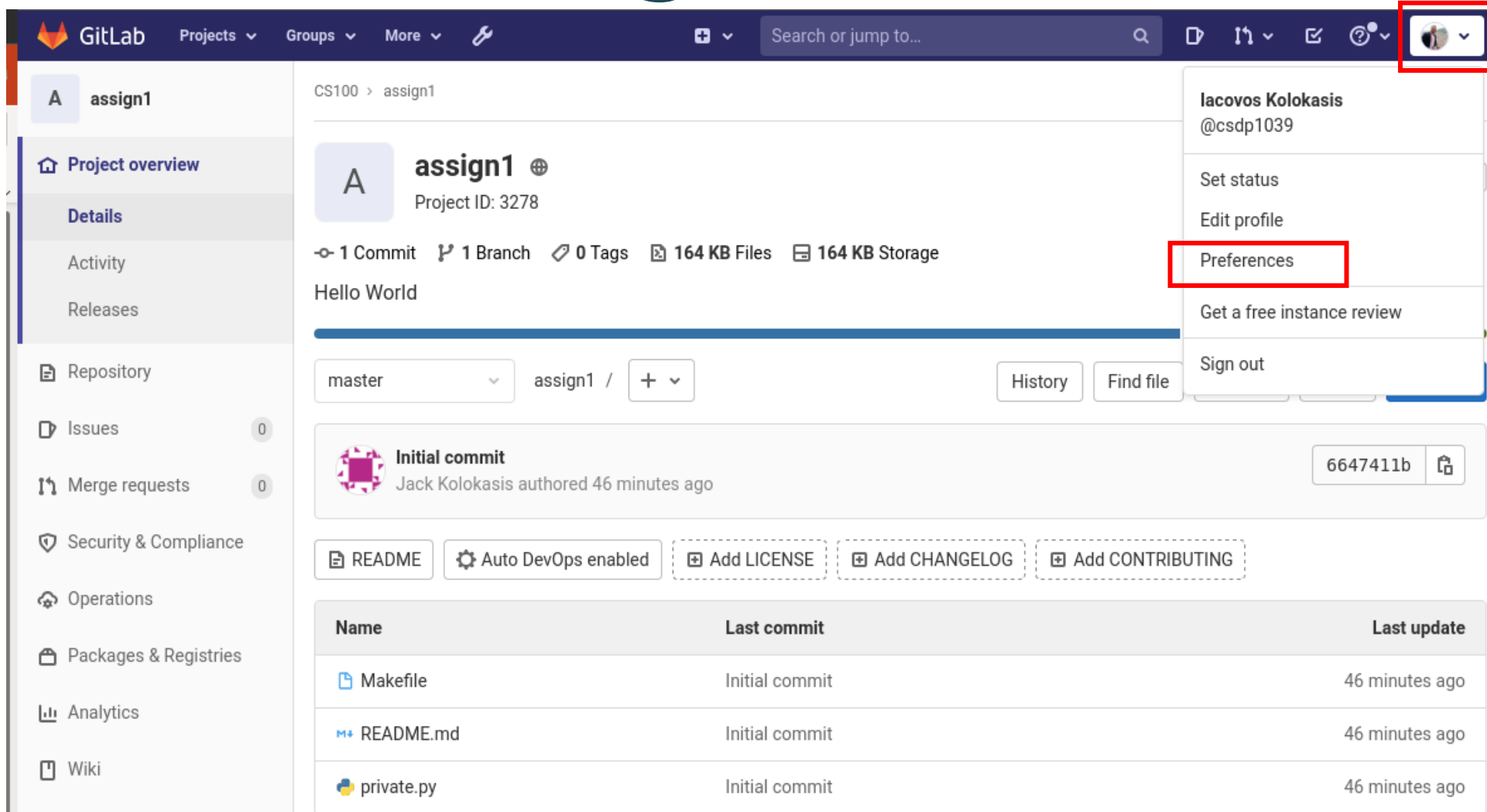
When you fork the repo you have to run `private.py` script to make change the visibility of the project as private and add TAs as members in the project.

The `private.py` script is written in [Python 3](#). To run `private.py` script, please have **Python 3.4 or higher**, **Git 1.8 or higher** and **python-gitlab** installed.

To install [python-gitlab](#), run: `pip3 install --upgrade python-gitlab --user`

```
python3 private.py -t <personal token> -p <project_id>
```

Generating an Access Token

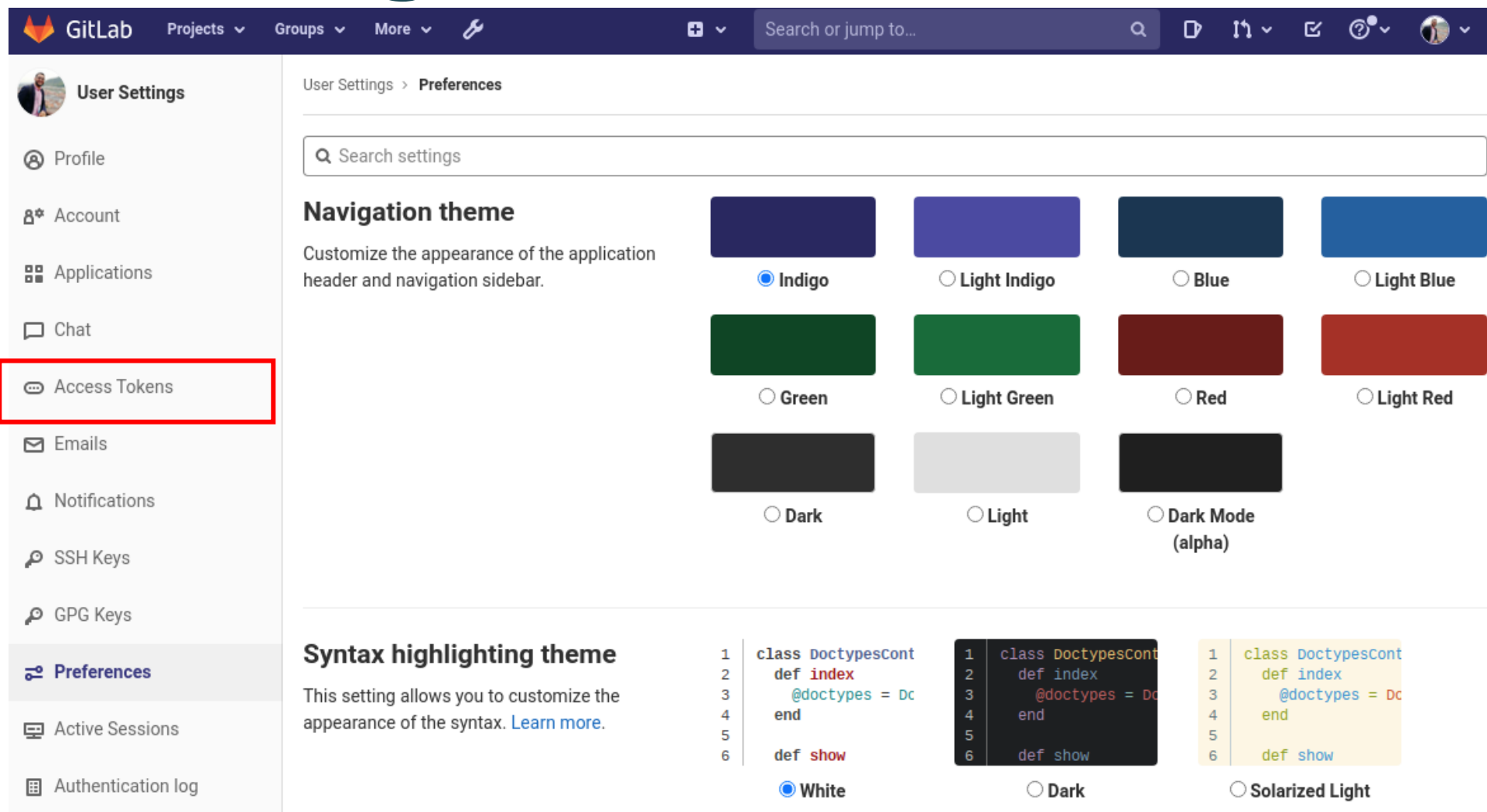


The screenshot shows the GitLab web interface. At the top, the navigation bar includes the GitLab logo, 'Projects', 'Groups', 'More', and a search bar. On the right side of the navigation bar, a user profile icon is highlighted with a red box. A dropdown menu is open from this icon, showing the user's name 'Iacovos Kolokasis' and email '@csdp1039'. The menu options are: 'Set status', 'Edit profile', 'Preferences' (highlighted with a red box), 'Get a free instance review', and 'Sign out'.

The main content area shows the 'assign1' project page. The project name 'assign1' is displayed with a globe icon and 'Project ID: 3278'. Below this, statistics show '1 Commit', '1 Branch', '0 Tags', '164 KB Files', and '164 KB Storage'. The commit message 'Hello World' is visible. A commit history section shows an 'Initial commit' by 'Jack Kolokasis' 46 minutes ago, with a commit hash '6647411b' and a copy icon. Below the commit history, there are buttons for 'README', 'Auto DevOps enabled', 'Add LICENSE', 'Add CHANGELOG', and 'Add CONTRIBUTING'.

Name	Last commit	Last update
Makefile	Initial commit	46 minutes ago
README.md	Initial commit	46 minutes ago
private.py	Initial commit	46 minutes ago

Generating an Access Token



User Settings

- Profile
- Account
- Applications
- Chat
- Access Tokens**
- Emails
- Notifications
- SSH Keys
- GPG Keys
- Preferences**
- Active Sessions
- Authentication log

User Settings > **Preferences**

Search settings

Navigation theme

Customize the appearance of the application header and navigation sidebar.

<input checked="" type="radio"/> Indigo	<input type="radio"/> Light Indigo	<input type="radio"/> Blue	<input type="radio"/> Light Blue
<input type="radio"/> Green	<input type="radio"/> Light Green	<input type="radio"/> Red	<input type="radio"/> Light Red
<input type="radio"/> Dark	<input type="radio"/> Light	<input type="radio"/> Dark Mode (alpha)	

Syntax highlighting theme

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

<pre>1 class DoctypesCont 2 def index 3 @doctypes = Dc 4 end 5 6 def show</pre> <input checked="" type="radio"/> White	<pre>1 class DoctypesCont 2 def index 3 @doctypes = Dc 4 end 5 6 def show</pre> <input type="radio"/> Dark	<pre>1 class DoctypesCont 2 def index 3 @doctypes = Dc 4 end 5 6 def show</pre> <input type="radio"/> Solarized Light
----------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------

Generating an Access Token

User Settings > Access Tokens

Q Search settings

Personal Access Tokens

You can generate a personal access token for each application you use that needs access to the GitLab API.

You can also use personal access tokens to authenticate against Git over HTTP. They are the only accepted password when you have Two-Factor Authentication (2FA) enabled.

Add a personal access token

Enter the name of your application, and we'll return a unique personal access token.

Name

Expires at

Scopes

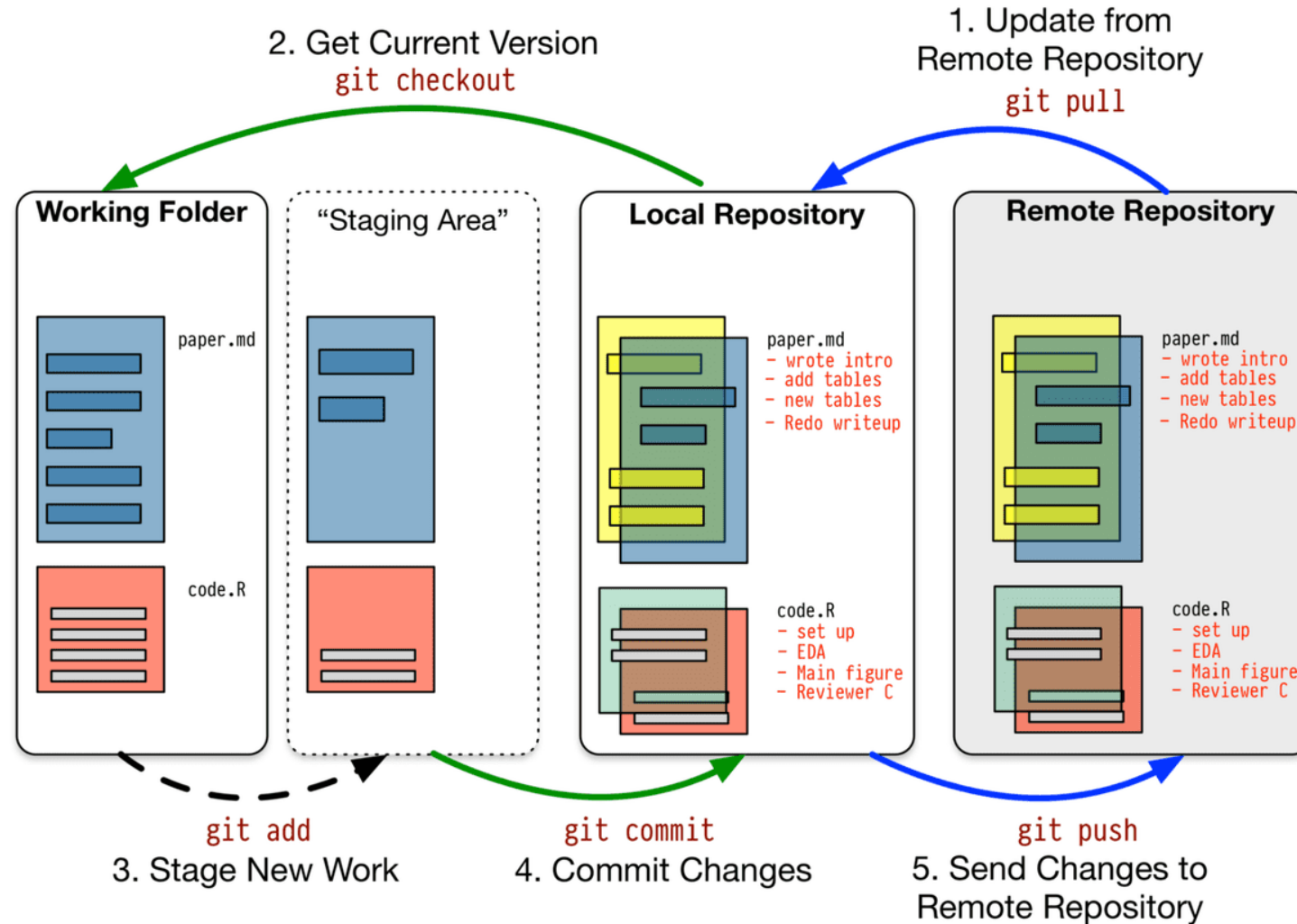
- ☐ **api**
Grants complete read/write access to the API, including all groups and projects, the container registry, and the package registry.
- ☐ **read_user**
Grants read-only access to the authenticated user's profile through the /user API endpoint, which includes username, public email, and full name. Also grants access to read-only API endpoints under /users.
- ☐ **read_api**
Grants read access to the API, including all groups and projects, the container registry, and the package registry.
- ☐ **read_repository**
Grants read-only access to repositories on private projects using Git-over-HTTP or the Repository Files API.
- ☐ **write_repository**
Grants read-write access to repositories on private projects using Git-over-HTTP (not using the API).
- ☐ **read_registry**
Grants read-only access to container registry images on private projects.
- ☐ **write_registry**
Grants write access to container registry images on private projects.
- ☐ **sudo**
Grants permission to perform API actions as any user in the system, when authenticated as an admin user.

Create personal access token

Add your name e.g., Iacovos Kolokasis

Select all the options

How does Git Work?



How to Use Git?

- To download a git repo run: `git clone repo-url`
- To setup your credentials in the local git repo run:
`git config user.name "Giorgos Xanthakis"`
`git config user.email "gxanth@csd.uoc.gr"`
- To stage a file run: `git add file`
- To commit staged files run: `git commit -m "Your Message"`
- To save your code in the server run: `git push`

Guidelines

- Commit often – at least one commit for each step
- Pay attention to your commit messages
- Describe precisely the purpose of the commit
- Commit only relevant files and modifications!!
- Push frequently (In case something goes wrong with your PC/laptop)

Thank you for your attention

Giorgos Xanthakis

Iacovos Kolokasis

University of Crete & ICS – FORTH

gxanth@csd.uoc.gr

kolokasis@csd.uoc.gr