

Git and Gitlab assignment

Main Task: GitHub & GitLab Collaboration & Workflow Setup

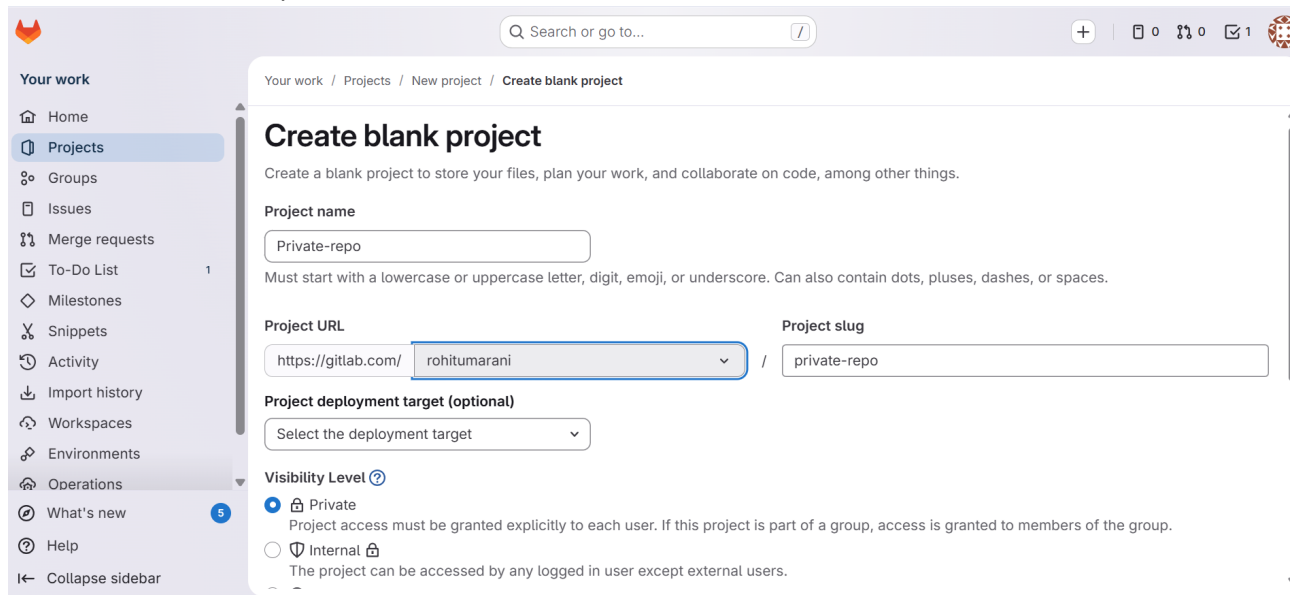
You are part of a DevOps team working on two different platforms – GitHub and GitLab. Your team wants to ensure smooth development, proper access control, and repository mirroring between the two platforms.

Part 2 : GitLab Tasks

Subtask 4: GitLab Repository Setup

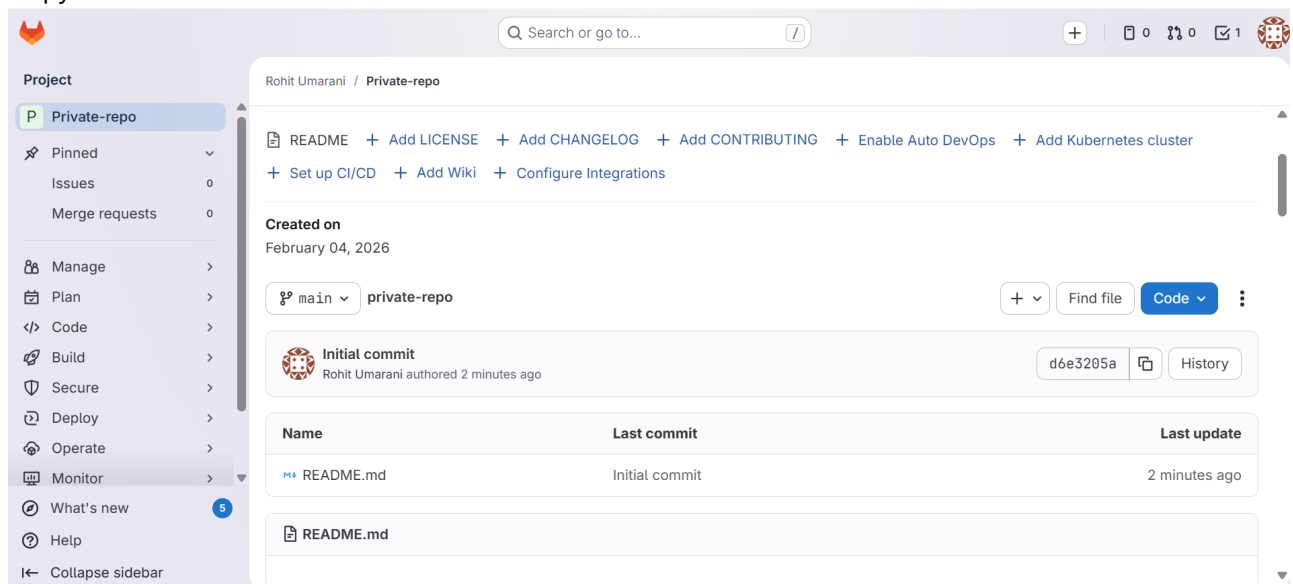
1. Create a private repository on GitLab.

- Created On Private Repo On GitLab

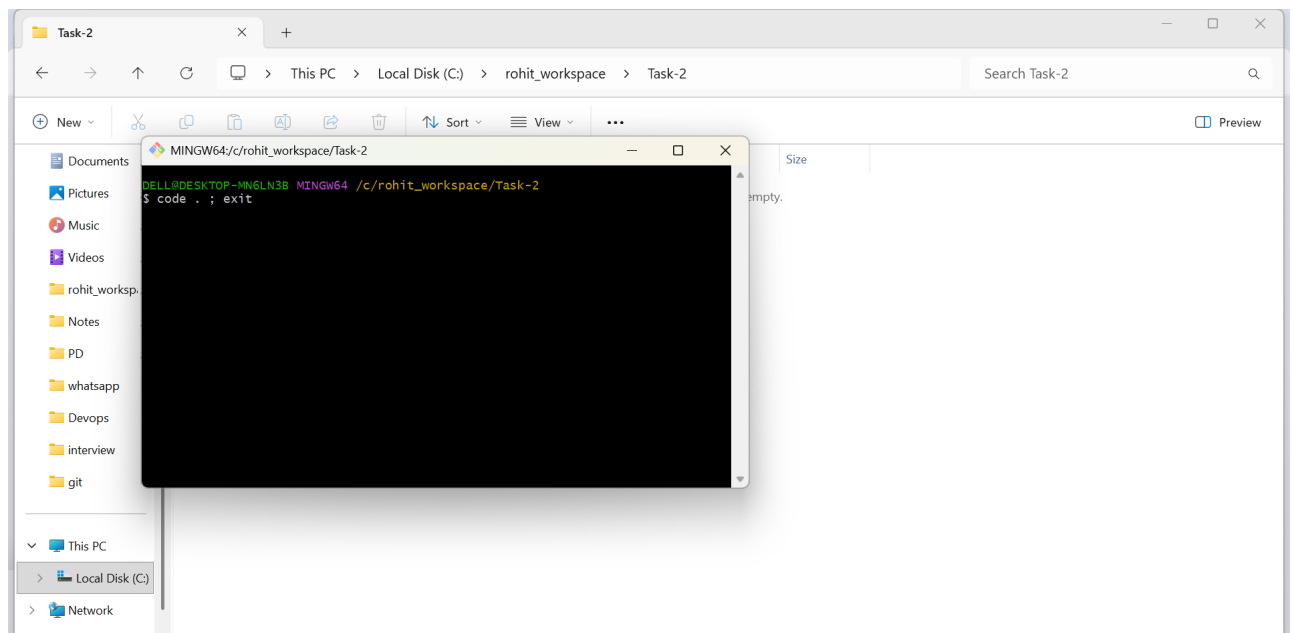


2. Clone it on your local machine using SSH (not HTTPS).

- Copy The URL SSH URL

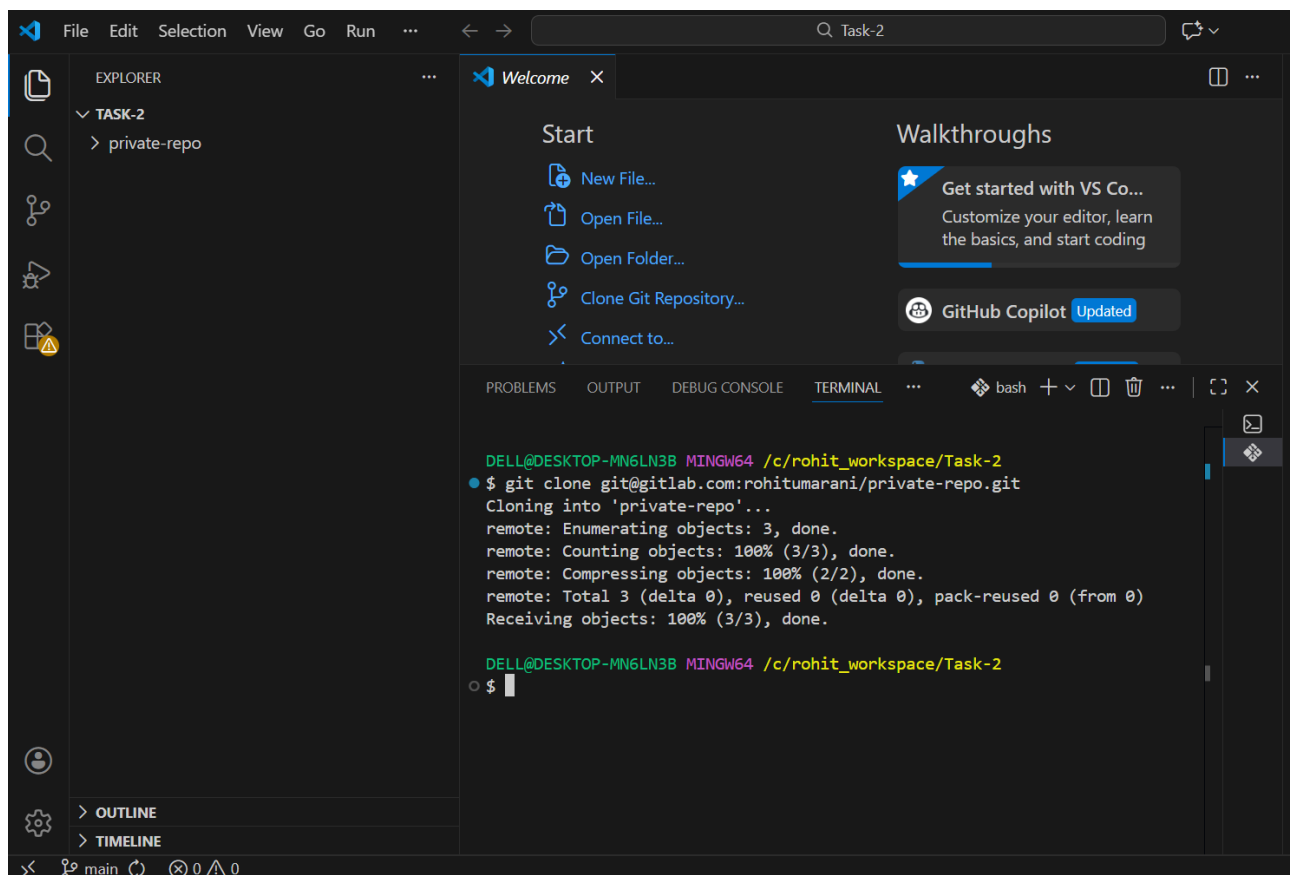


- Create One Folder Into Local Machine

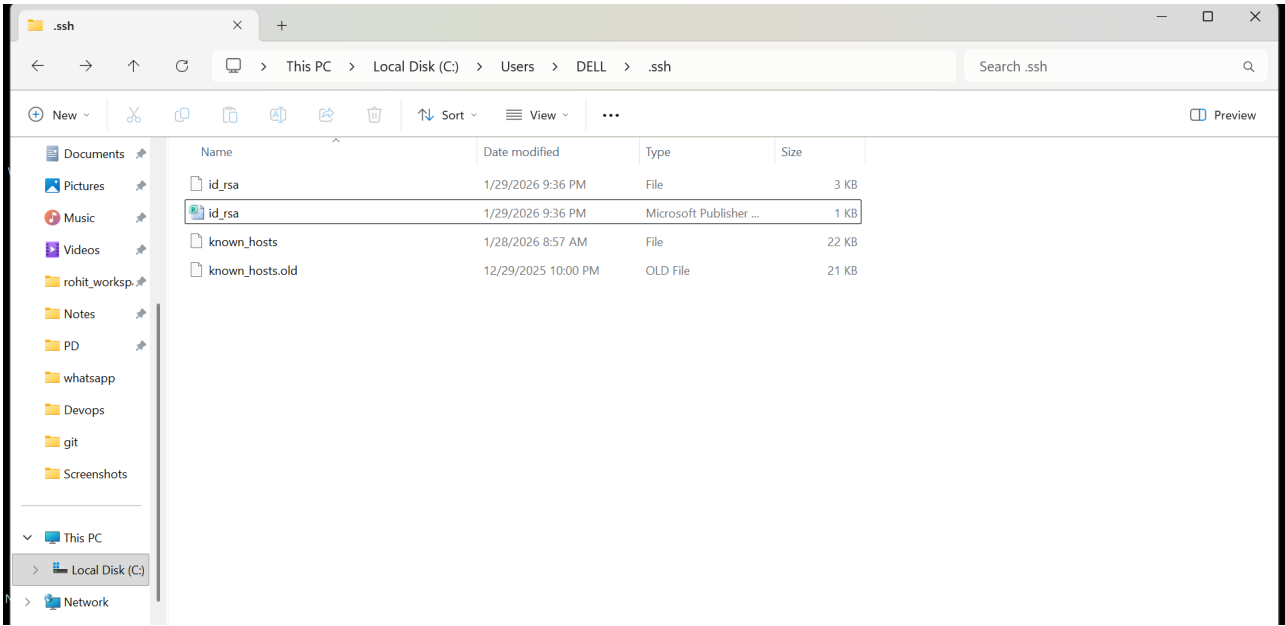


- Use Command For Clone Private Repo To Local Machine

```
git clone git@gitlab.com:rohitumarani/private-repo.git
```



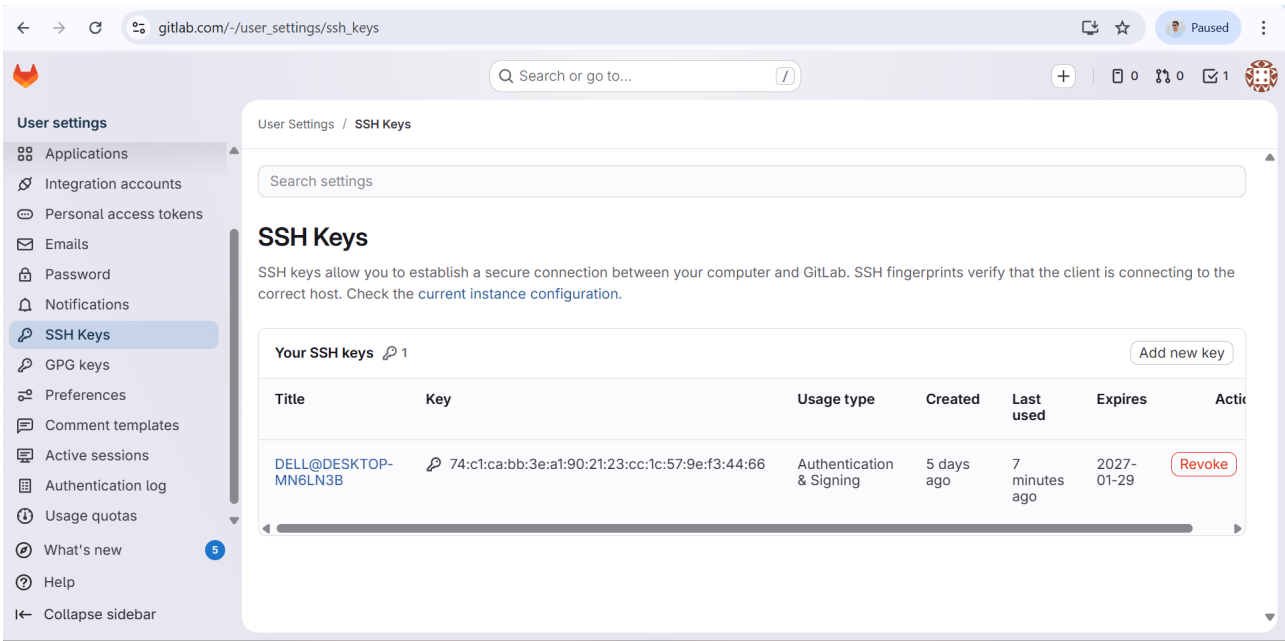
- Already Created SSH Key



How to check the Path:

```
C:\Users\DELL\.ssh
```

- SSH Key On GitLab



3. Create a simple project structure (e.g., src/app.py , docs/guide.md).

- Created Src Folder inside that app.py file

The screenshot shows the Visual Studio Code interface. In the Explorer panel on the left, the file structure is: TASK-2 > private-repo > src. The 'src' folder is selected, and it contains 'app.py' and 'README.md'. The main editor area shows the 'Welcome' page with options like 'New File...', 'Open File...', and 'Open Folder...'. The terminal at the bottom shows the following commands and output:

```
DELL@DESKTOP-MN6LN3B MINGW64 /c/rohit_workspace/Task-2
• $ cd private-repo/

DELL@DESKTOP-MN6LN3B MINGW64 /c/rohit_workspace/Task-2/private-repo (main)
• $ mkdir src

DELL@DESKTOP-MN6LN3B MINGW64 /c/rohit_workspace/Task-2/private-repo (main)
• $ cd src

DELL@DESKTOP-MN6LN3B MINGW64 /c/rohit_workspace/Task-2/private-repo/src (main)
• $ touch app.py

DELL@DESKTOP-MN6LN3B MINGW64 /c/rohit_workspace/Task-2/private-repo/src (main)
• $
```

- Created docs Folder inside that guide.md file

The screenshot shows the Visual Studio Code interface. In the Explorer panel on the left, the file structure is: TASK-2 > private-repo > docs. The 'docs' folder is selected, and it contains 'guide.md' and 'src'. The main editor area shows a large 'X' watermark. The terminal at the bottom shows the following commands and output:

```
DELL@DESKTOP-MN6LN3B MINGW64 /c/rohit_workspace/Task-2/private-repo (main)
• $ mkdir docs

DELL@DESKTOP-MN6LN3B MINGW64 /c/rohit_workspace/Task-2/private-repo (main)
• $ cd docs

DELL@DESKTOP-MN6LN3B MINGW64 /c/rohit_workspace/Task-2/private-repo/docs (main)
• $ touch guide.md

DELL@DESKTOP-MN6LN3B MINGW64 /c/rohit_workspace/Task-2/private-repo/docs (main)
• $
```

Subtask 5: Repository Mirroring

Create a mirror setup:

- Set the GitHub private repo as the mirror of your GitLab repo.

- Created One New Repo On Github For Mirroring

Owner * rohitumarani / Repository name * Github-mirror-repo-task

Great repository names are short and memorable. How about [stunning-train](#)?

Description

0 / 350 characters

2 Configuration

Choose visibility * Public

Add README Off

Add .gitignore No .gitignore

Add license No license

- Add This Github Repository to the GitLab Repository

Repository - Settings - Rohit Um x rohitumarani/Github-mirror-rep x +

gitlab.com/rohitumarani/private-repo/-/settings/repository#js-push-remote-settings

Search or go to...

Project

- Analyze
- Settings
- General
- Integrations
- Webhooks
- Access tokens
- Repository
- Merge requests
- CI/CD
- Packages and registries
- Monitor
- Usage quotas

What's new 5

Help

Collapse sidebar

Rohit Umarani / Private-repo / Repository settings

Add new mirror repository

Git repository URL

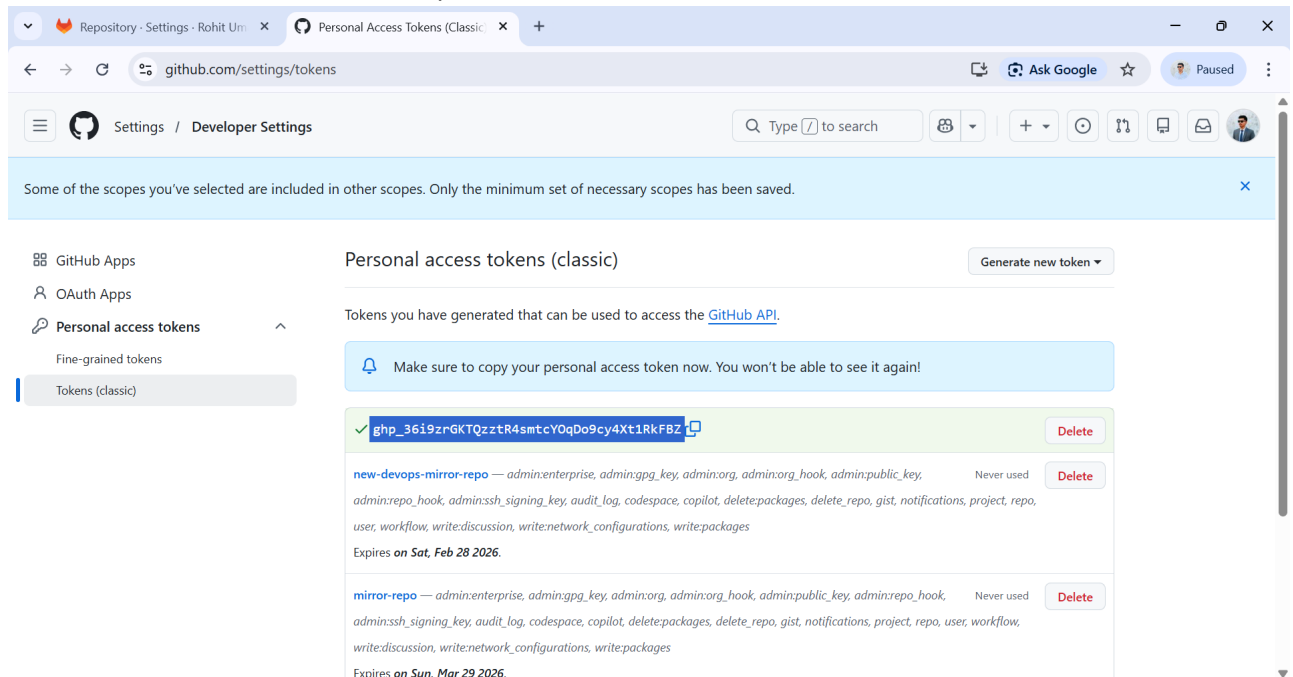
`https://github.com/rohitumarani/Github-mirror-repo-task.git`

- The repository must be accessible over `http://`, `https://`, `ssh://` or `git://`.
- When using the `http://` or `https://` protocols, please provide the exact URL to the repository. HTTP redirects will not be followed.
- Do not include the username in the URL, use the username field below if required:
`https://gitlab.company.com/group/project.git`
- When using the `ssh://` protocol, please use the following format: `ssh://username@example.com/group/project.git`.
- The update action will time out after 180 minutes. For big repositories, use a clone/push combination.
- Pull mirrors will only create LFS objects if LFS is [enabled for the project](#).
- Push mirrors will only sync LFS objects if LFS is [enabled for the project](#).
- Push mirrors will **not** sync LFS objects over SSH.
- In case of pull mirroring, your user will be the author of all events in the activity feed that are the result of an update, like new branches being created or new commits being pushed to existing branches.

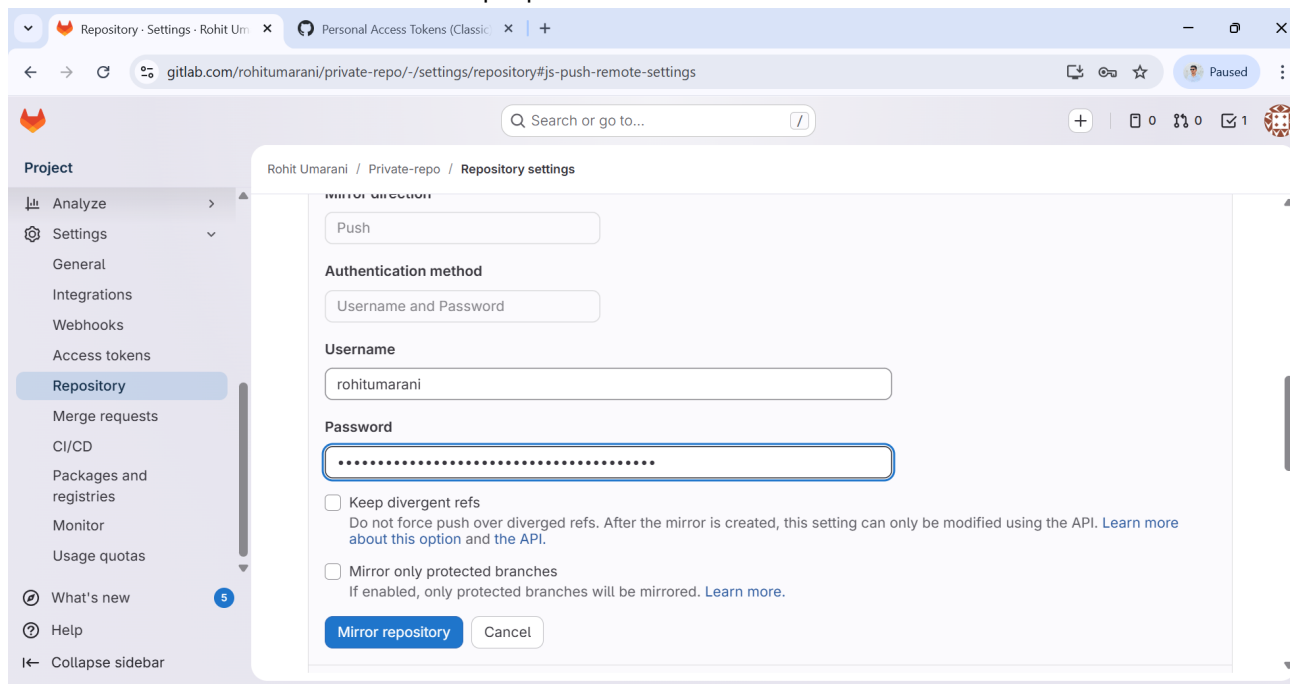
Mirror direction

Push

- Create Personal Access Token. It Required For Password Authentication

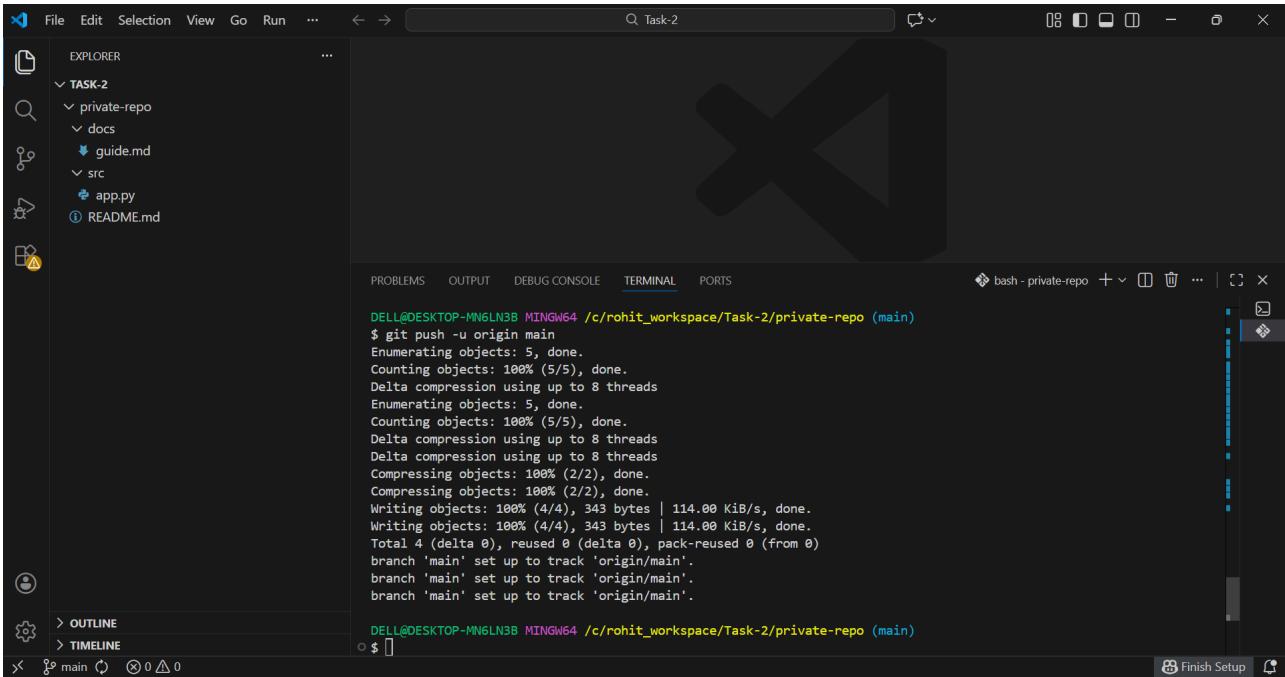


- After That Paste That Token And Setup Up GitLab and GitHub



2. Push some changes to GitLab and verify if the changes reflect in GitHub automatically.

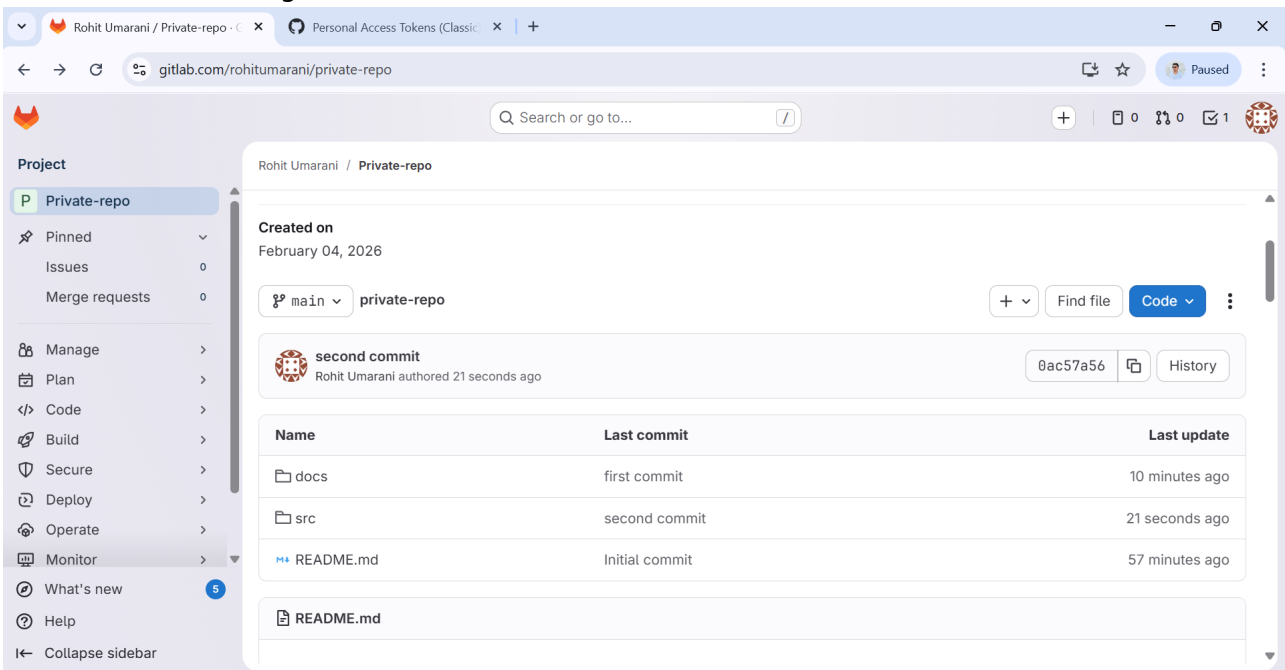
- Push the Changes



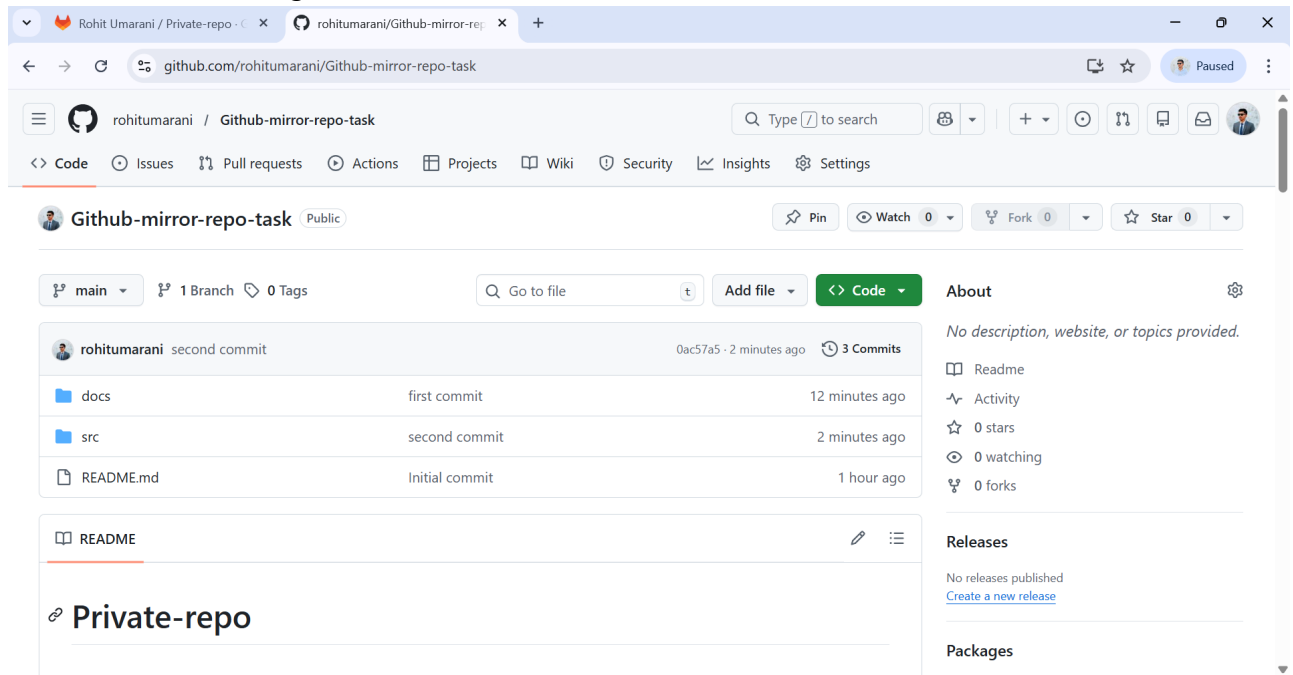
Use Commdnd For Push

```
git push -u origin main
```

- Check On GitLab Changes



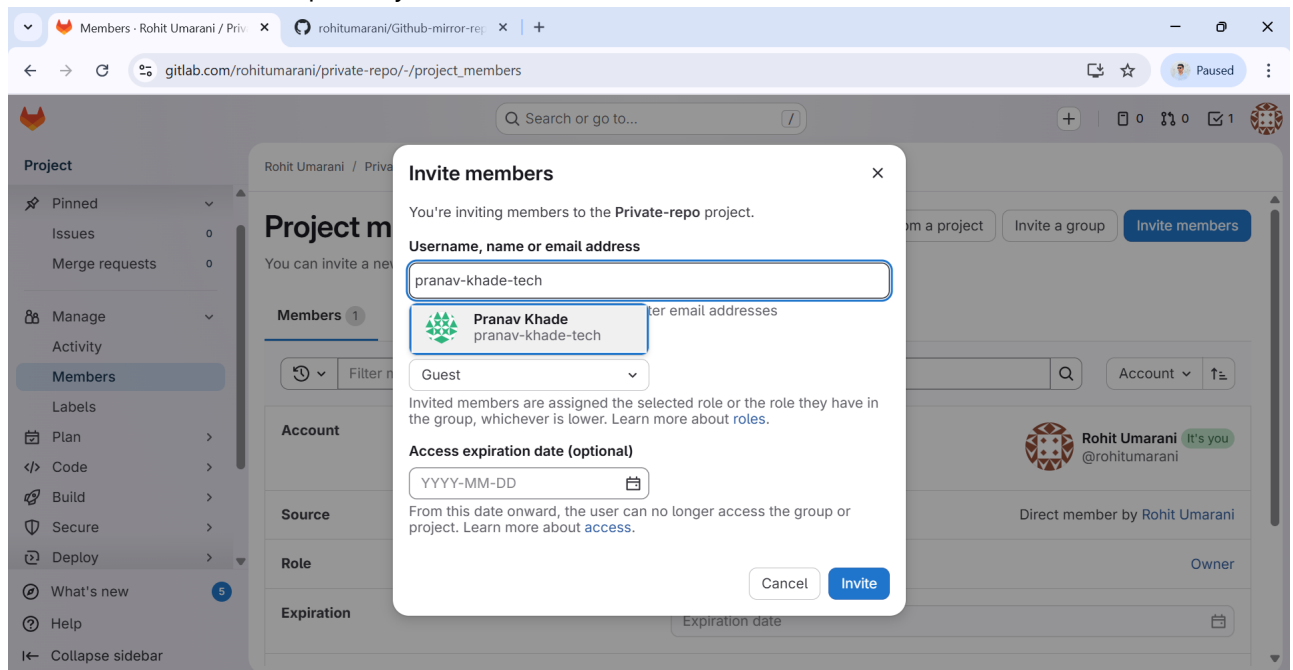
- Check On Github Changes



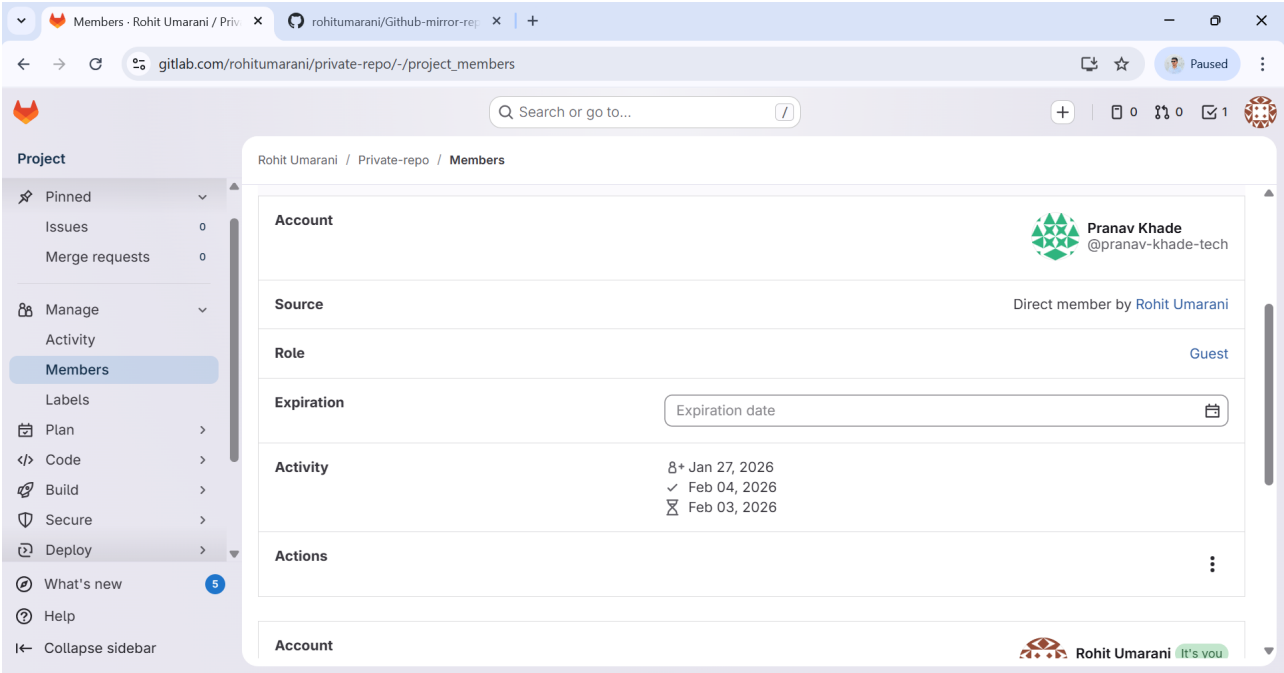
Subtask 6: Access Control

Invite your friend to the GitLab private repository:

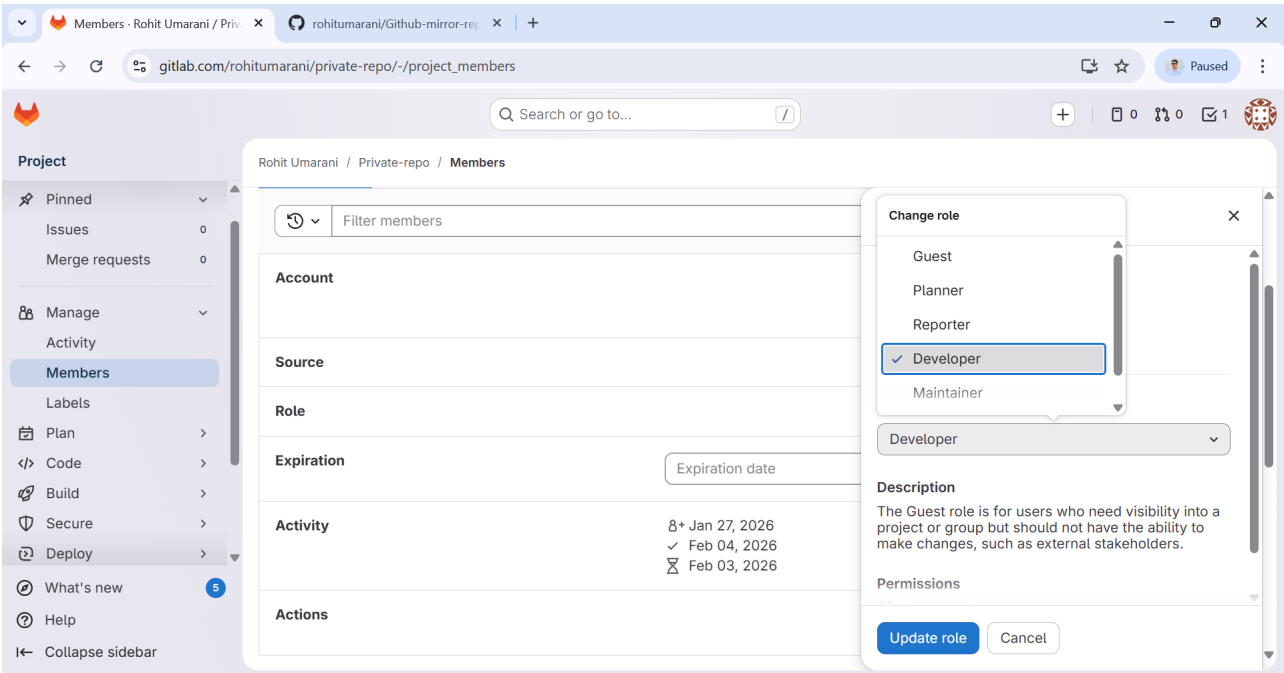
- Invite Member to the Repository



- Assign them the Guest role initially, observe the access.



- Then change their role to Developer, and let them push one file.



Role is Change To the Devolper

