DevOps Lab - Mukesh T P

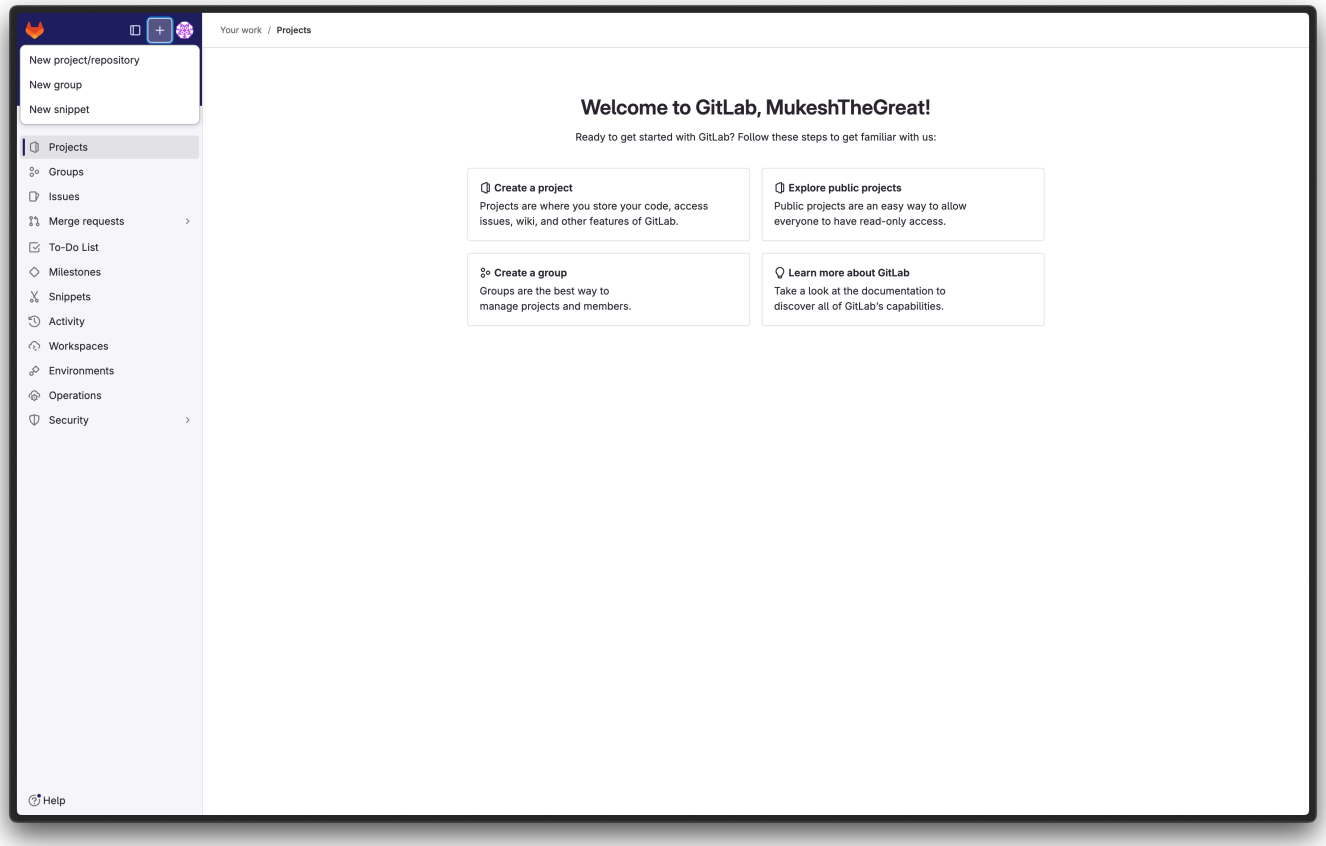
Exercise 2

Ex. No. 1	Simple Linear Regression
03.01.2024	

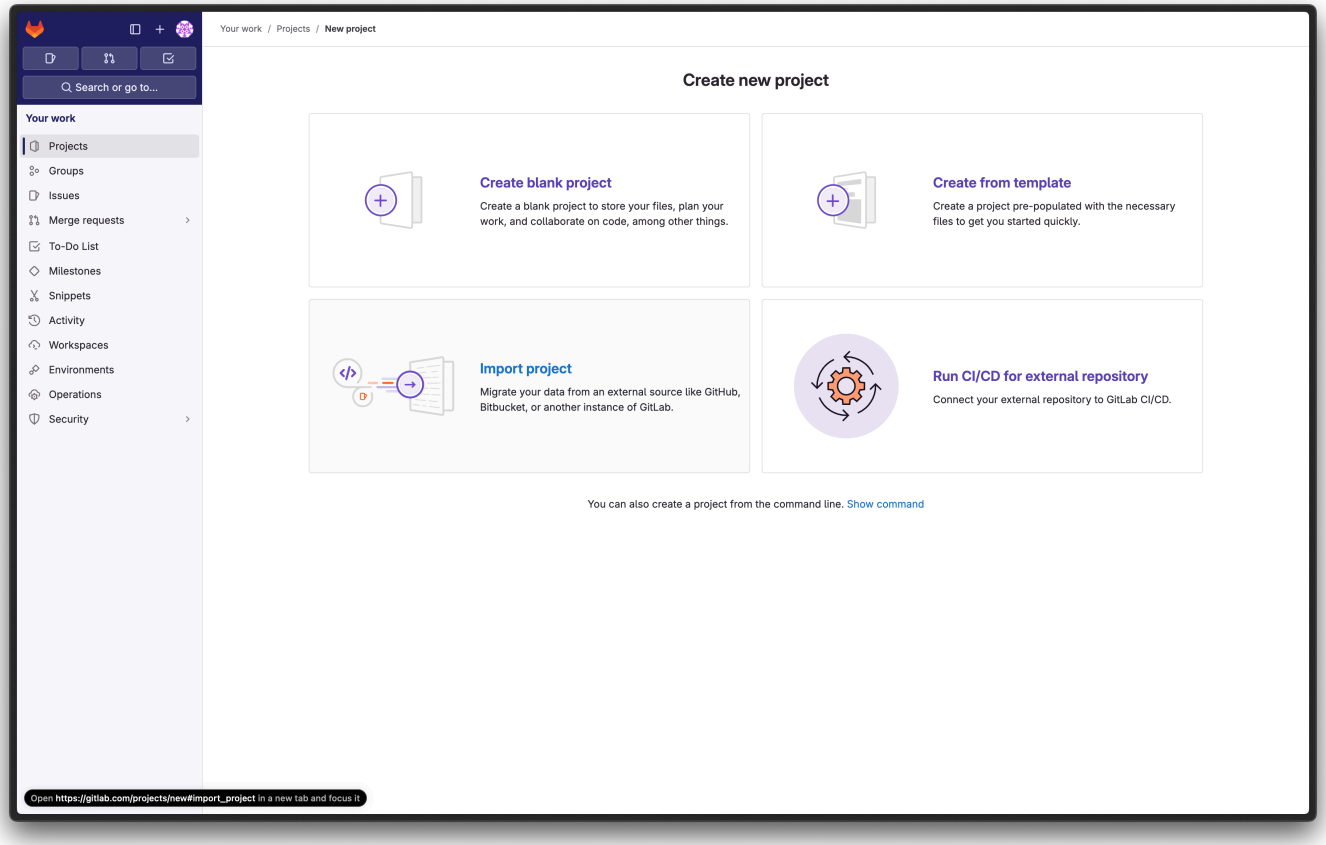
Implement GitLab Operations using Git

1. Creating a Repository

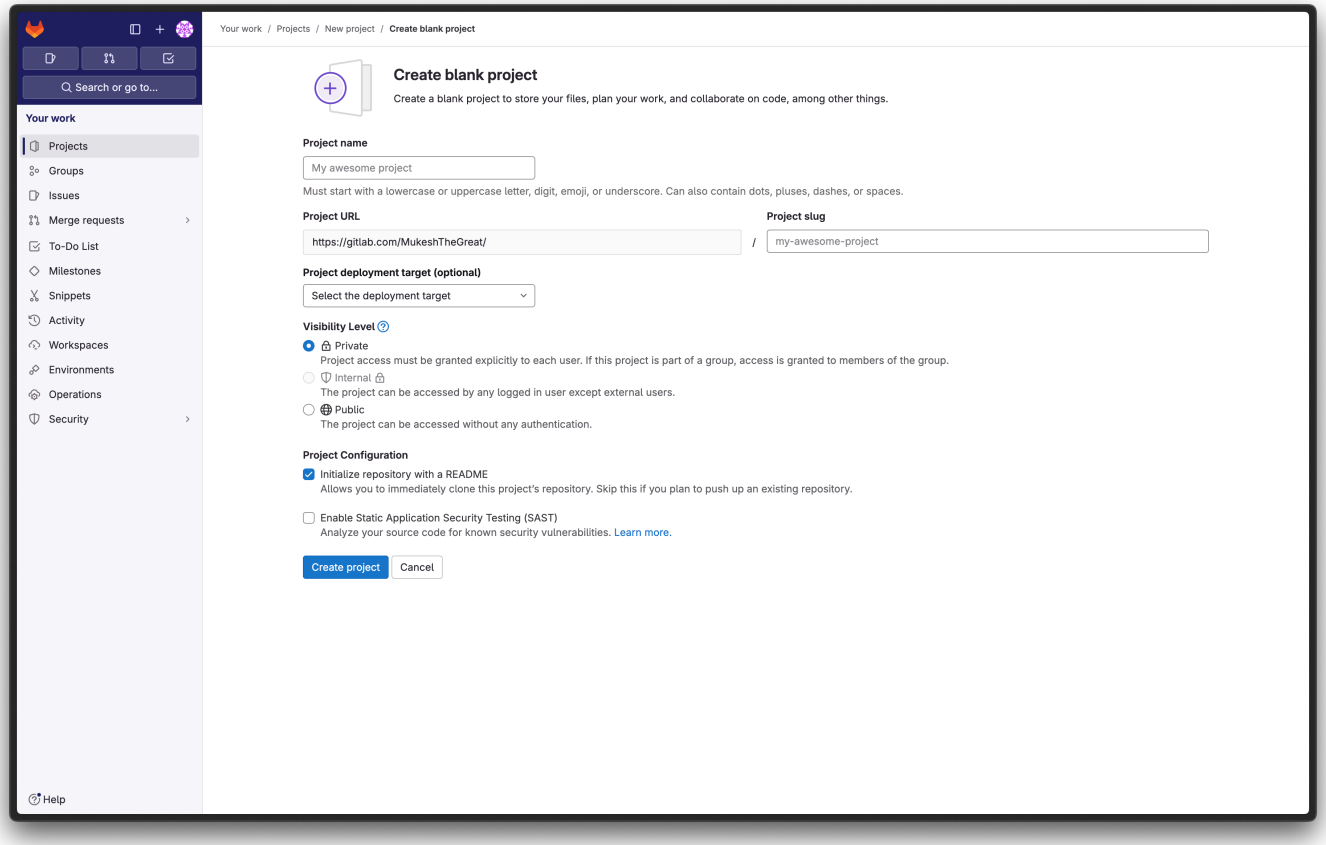
1. Go to your GitLab account.



2. Click on the `New project` button.

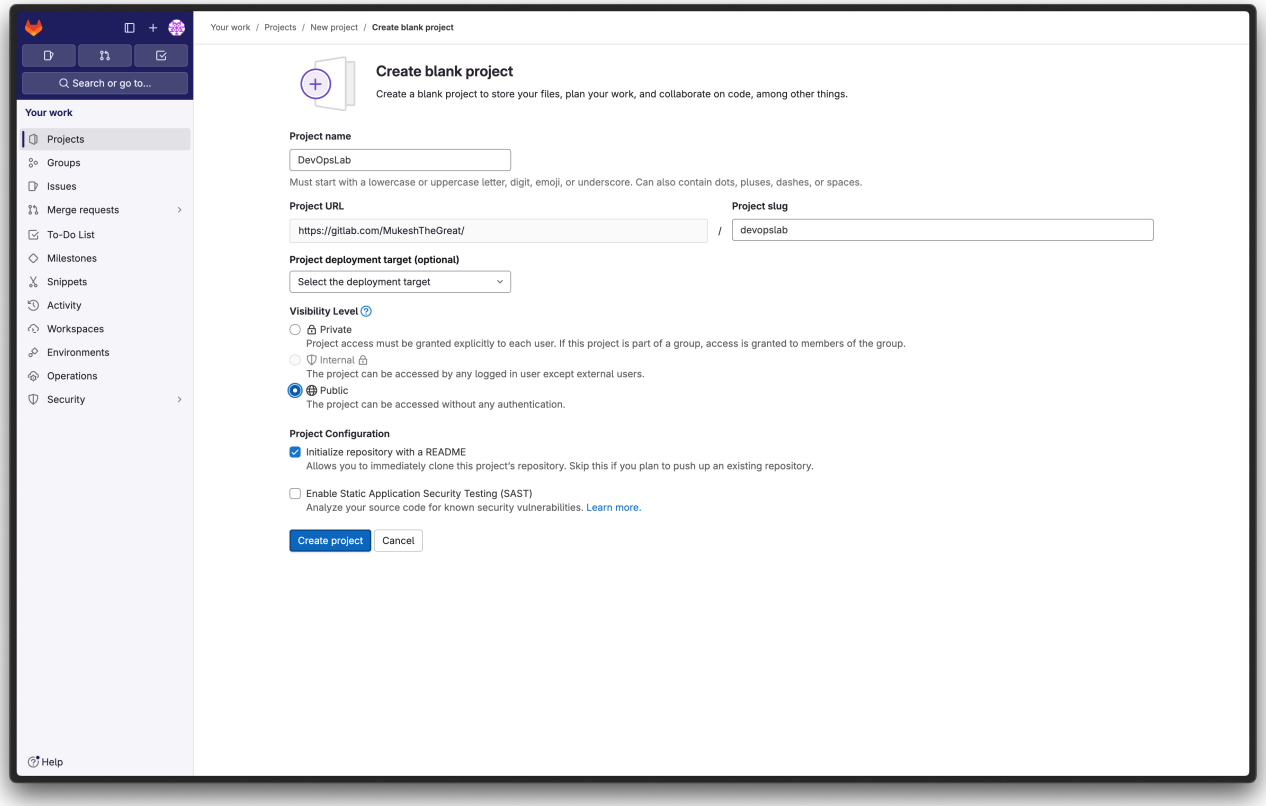


3. Select `Create blank project`.

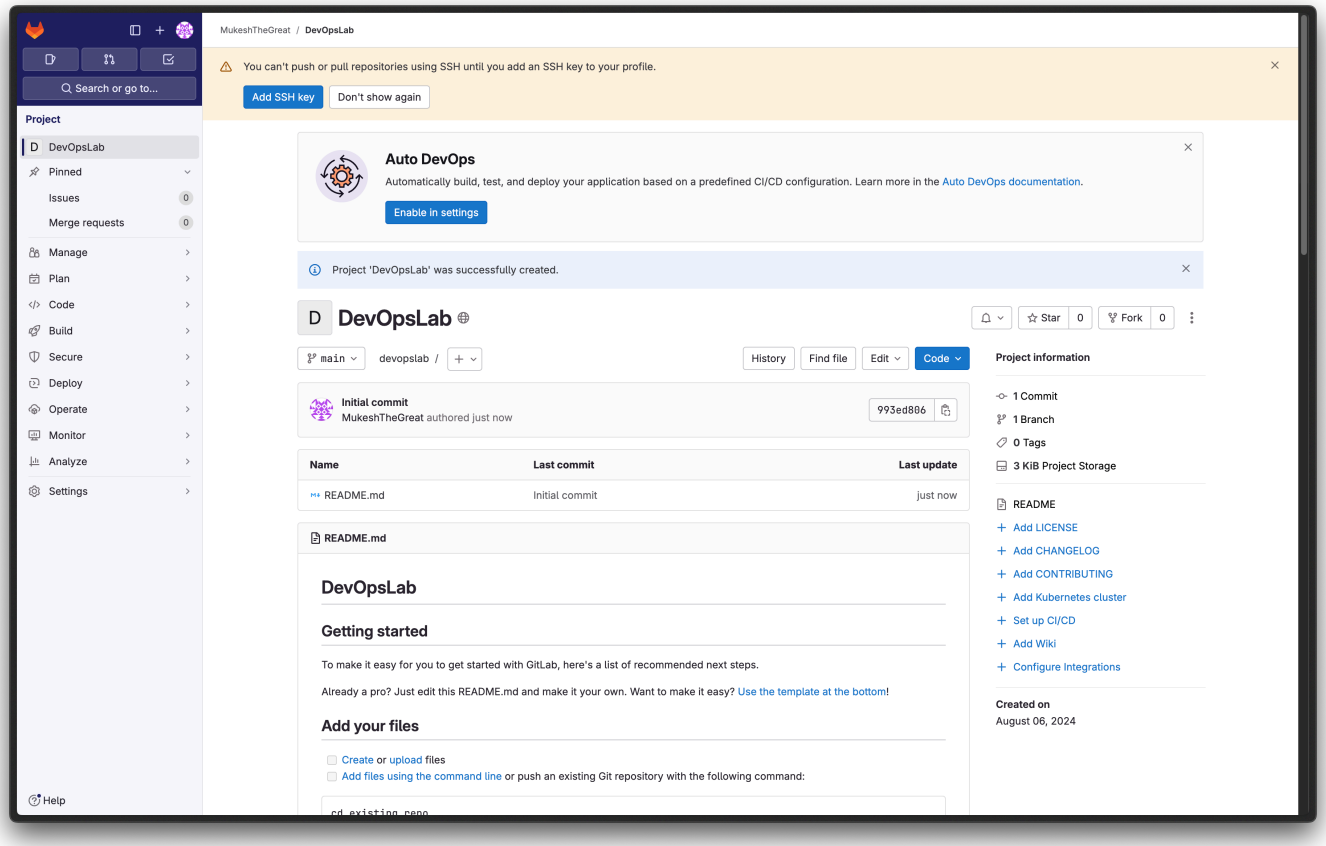


4. Fill in the project details:
   - Project name: `DevOpsLab`

- Project slug: `devopslab` (auto-filled)
- Visibility level: Choose between Private, Internal, or Public.

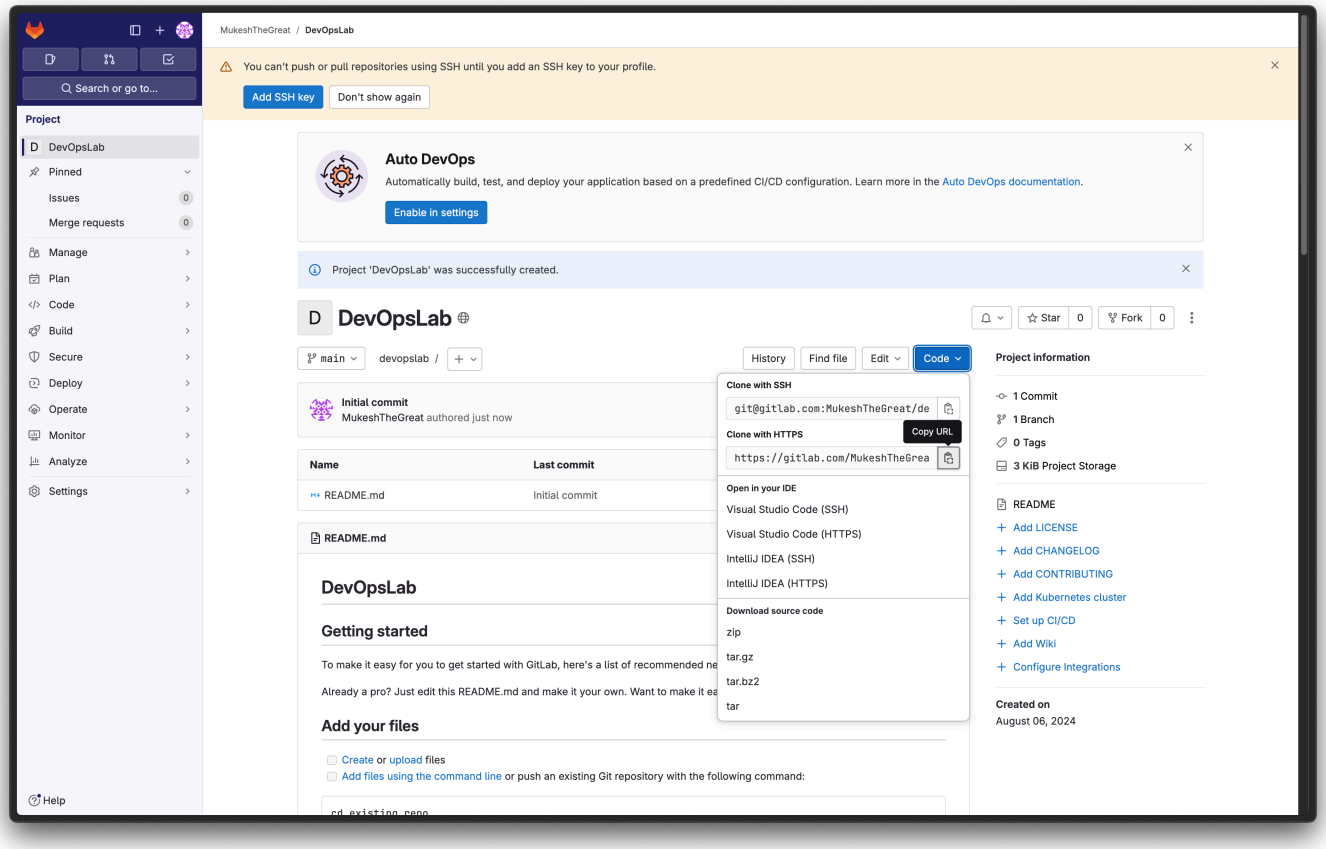


5. Click on the `Create project` button.

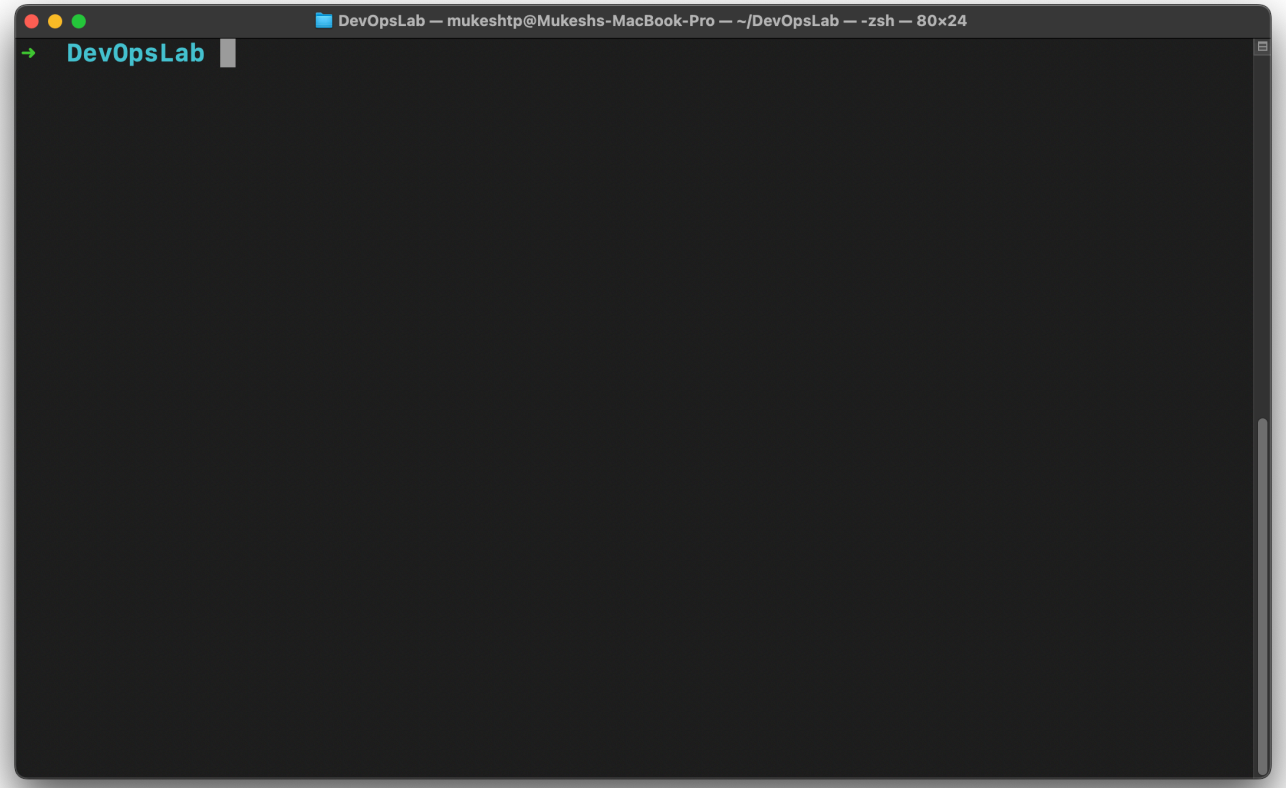


2. Cloning a Repository

1. In your GitLab repository, find the `Clone` button and copy the repository URL.

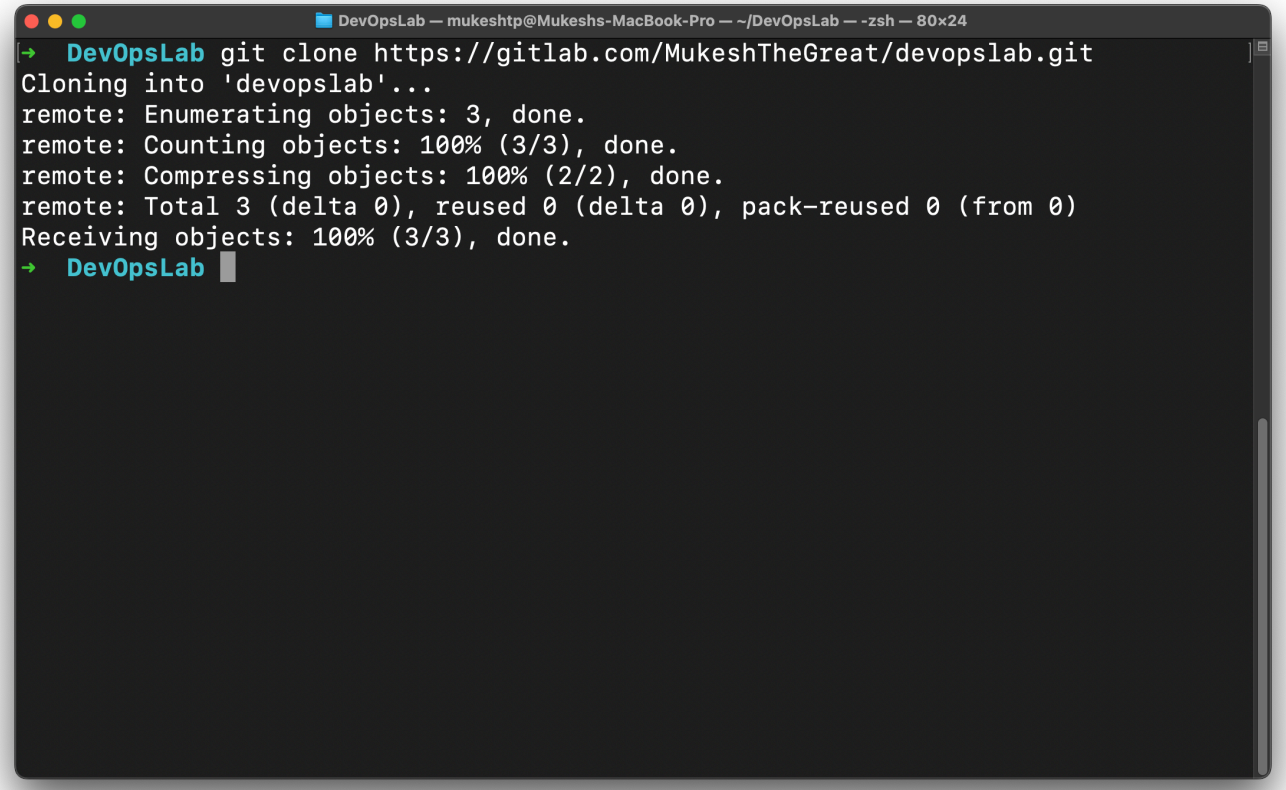


2. Open your terminal.



3. Clone the repository using the following command:

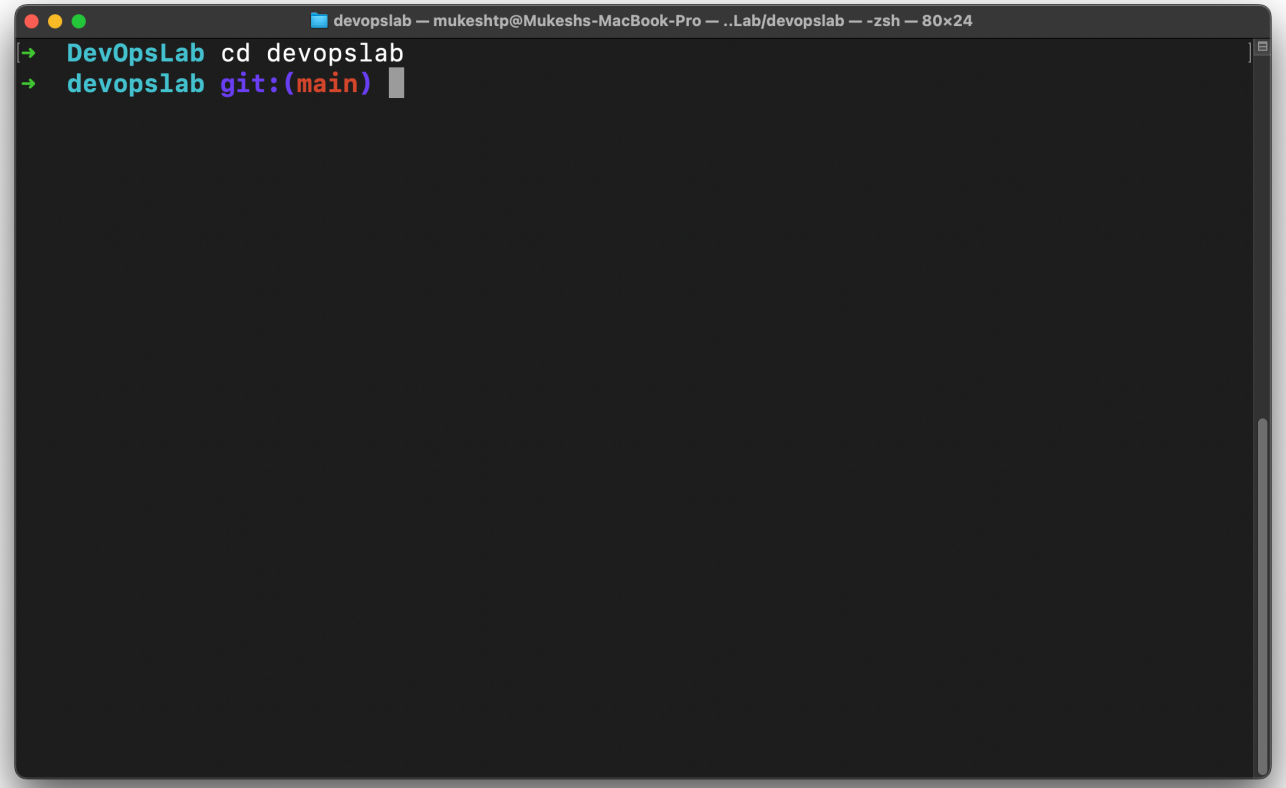
```
git clone https://gitlab.com/MukeshTheGreat/devopslab.git
```



3. Making Changes and Creating a Branch

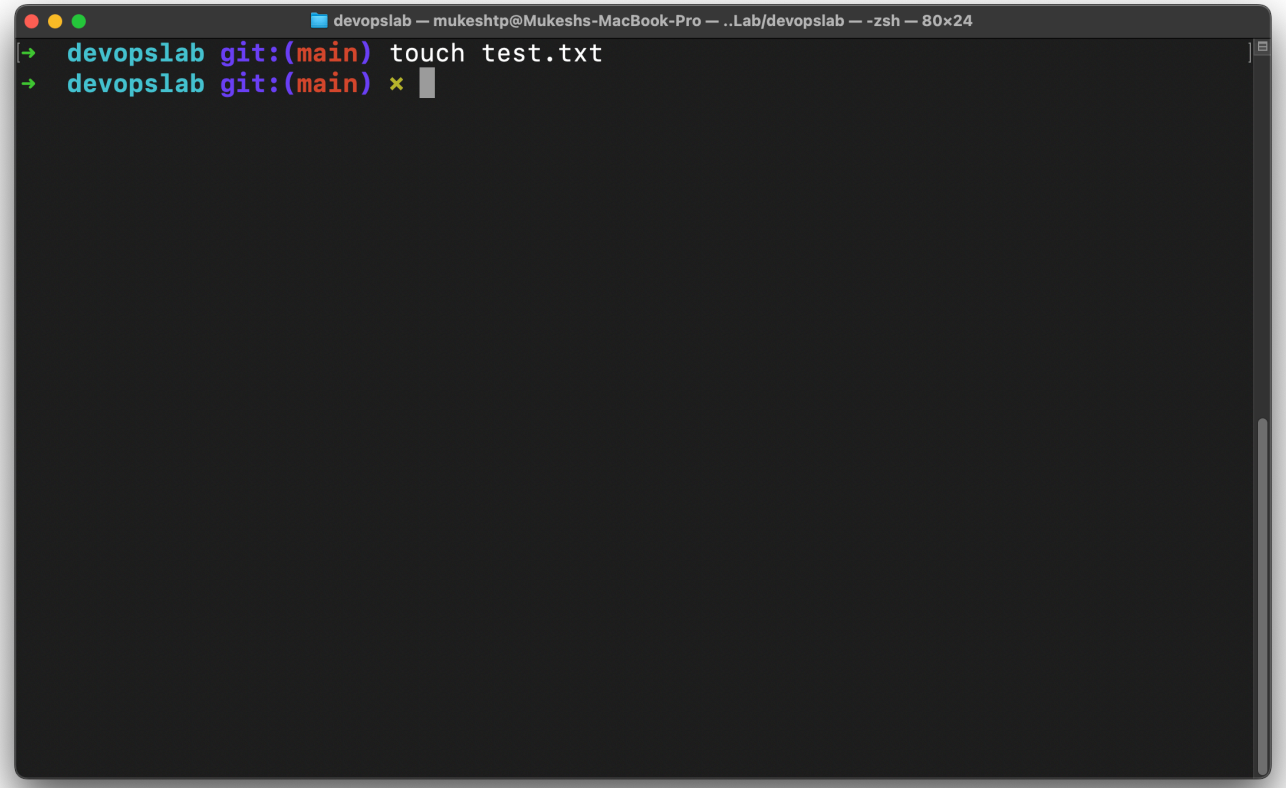
1. Navigate into the cloned repository:

```
cd example-repo
```



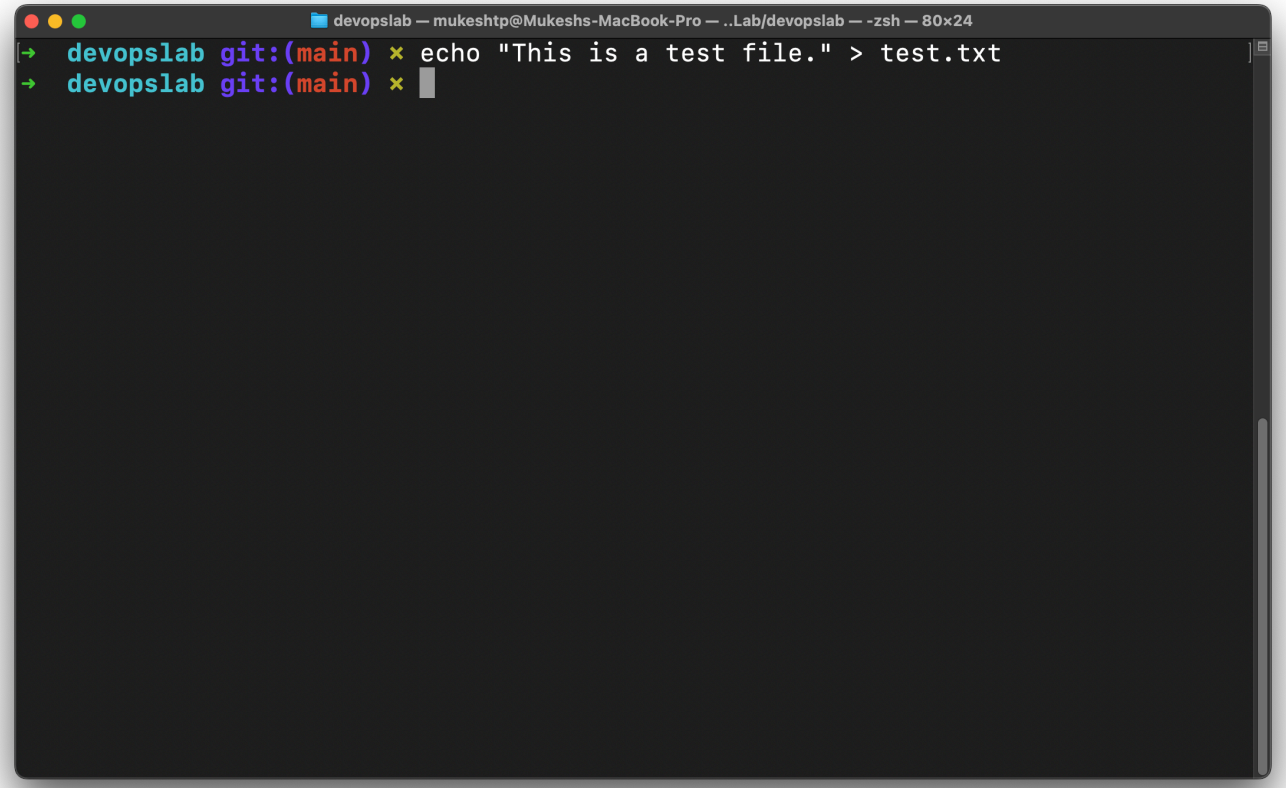
2. Create a new text file named `test.txt` :

```
touch test.txt
```



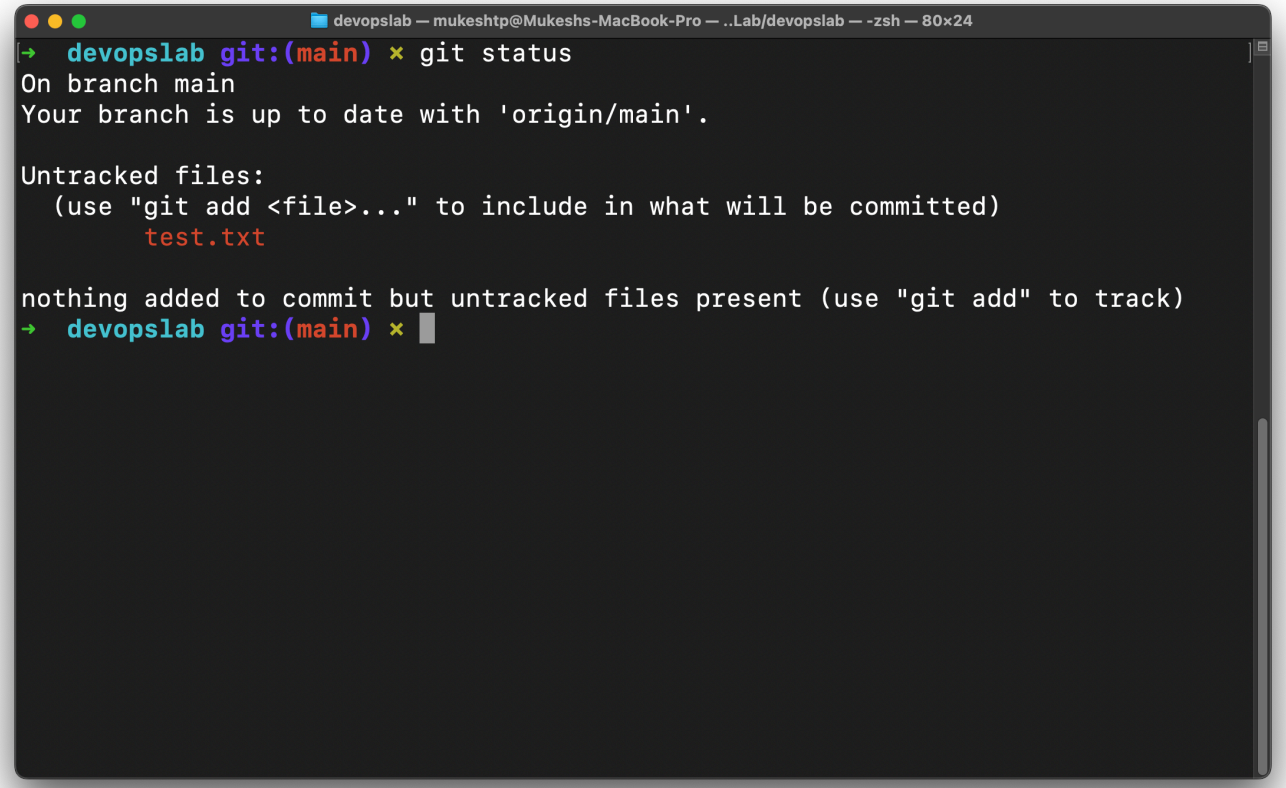
3. Add some content to the `test.txt` file.

```
echo "This is a test file." › test.txt
```



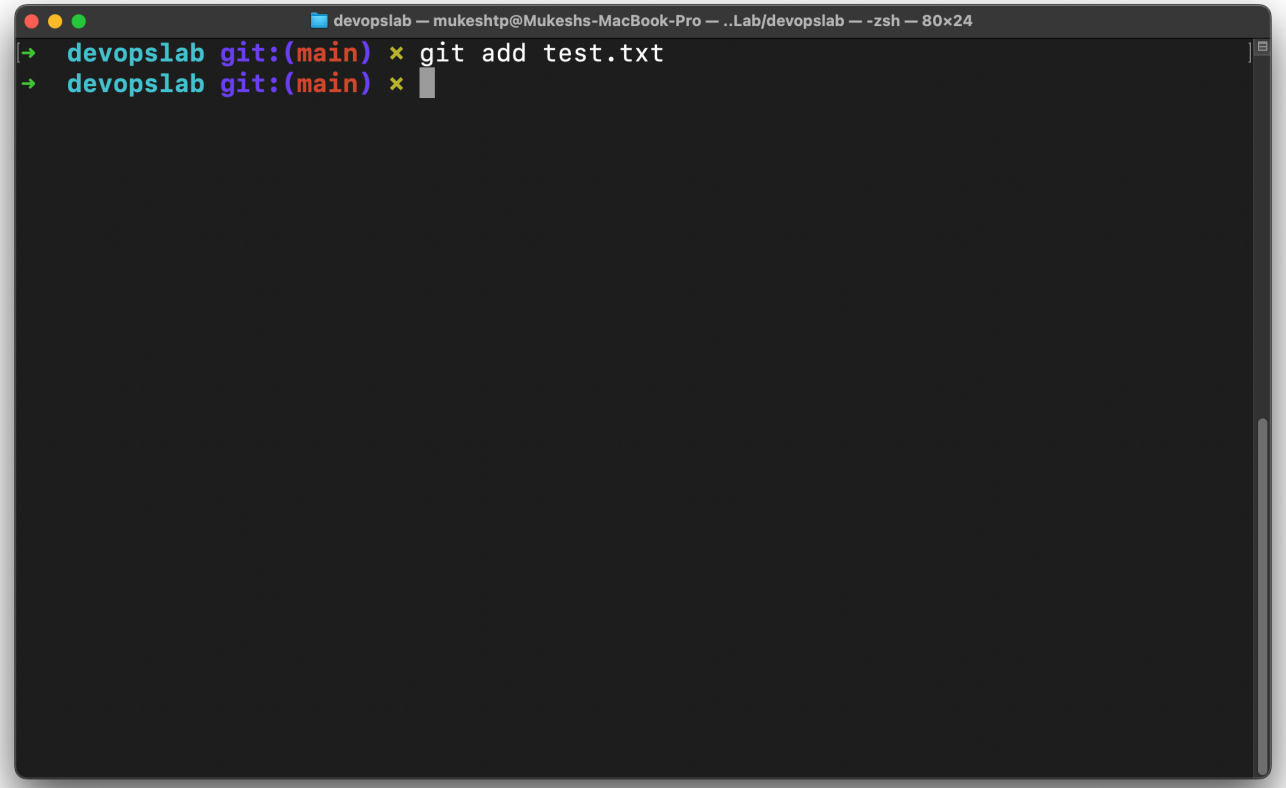
4. Check the status of the repository:

```
git status
```



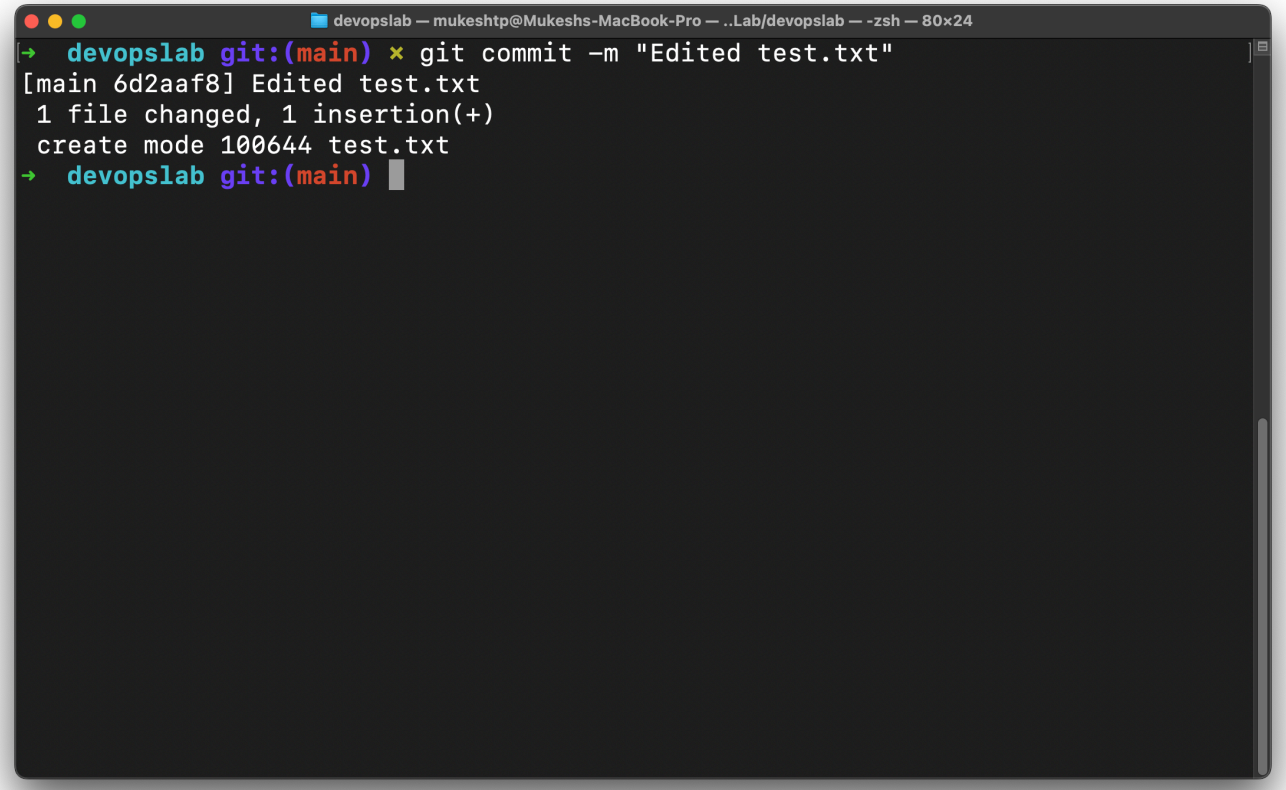
5. Stage the changes for commit:

```
git add test.txt
```



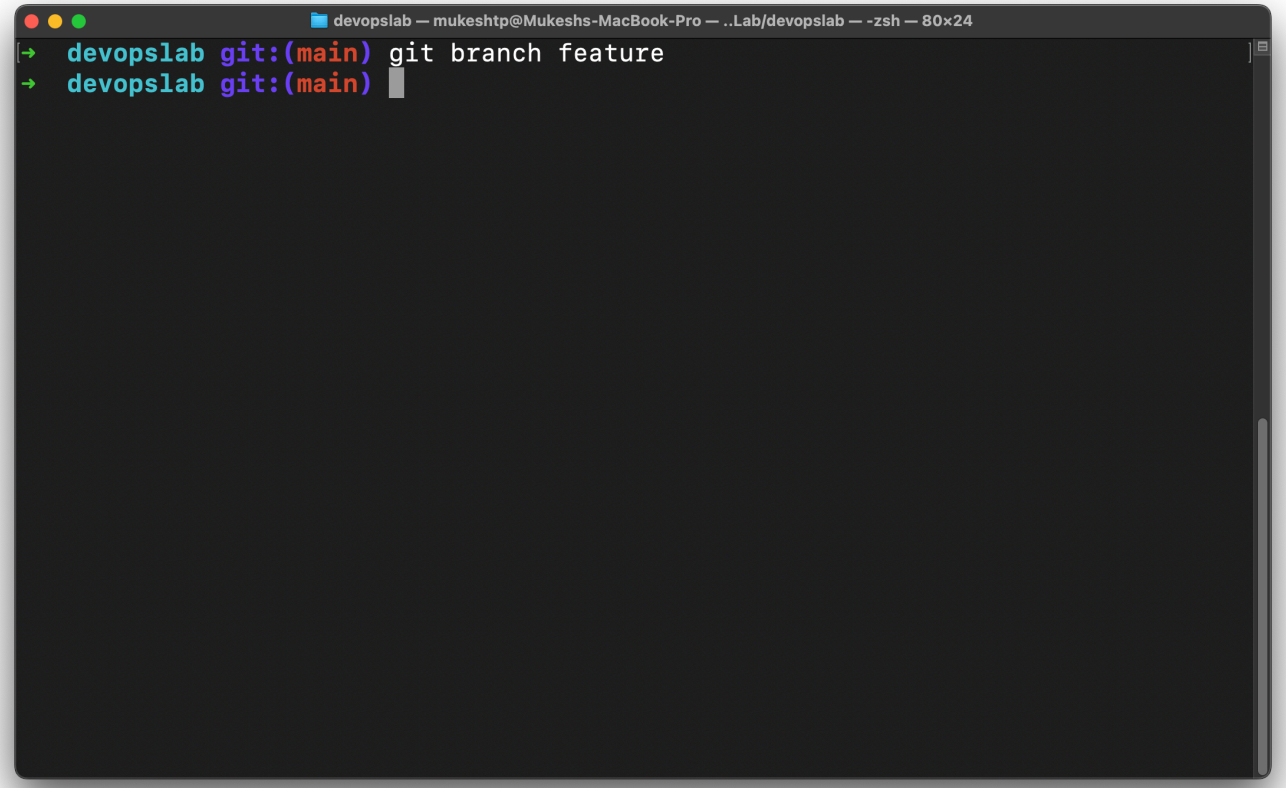
6. Commit the changes with a descriptive message:

```
git commit -m "Edited test.txt"
```



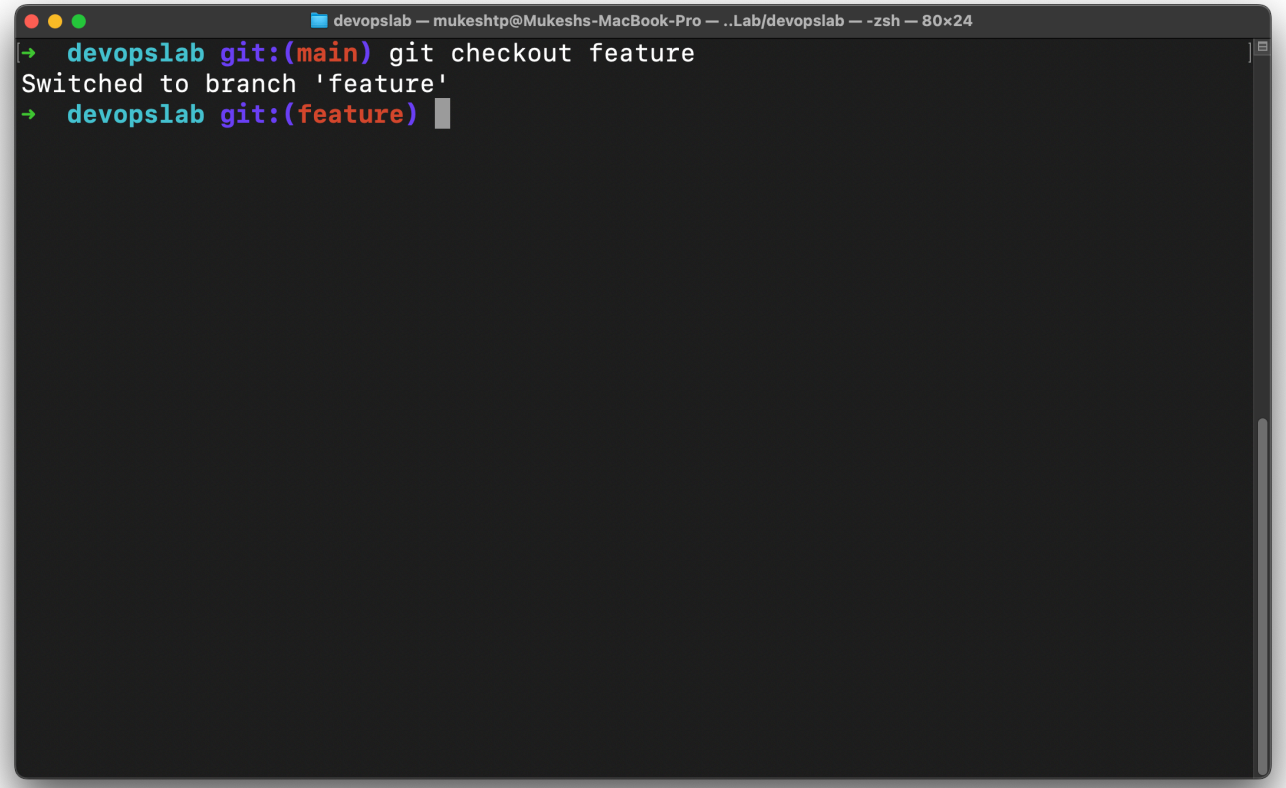
7. Create a new branch named `feature`:

```
git branch feature
```



8. Switch to the `feature` branch:

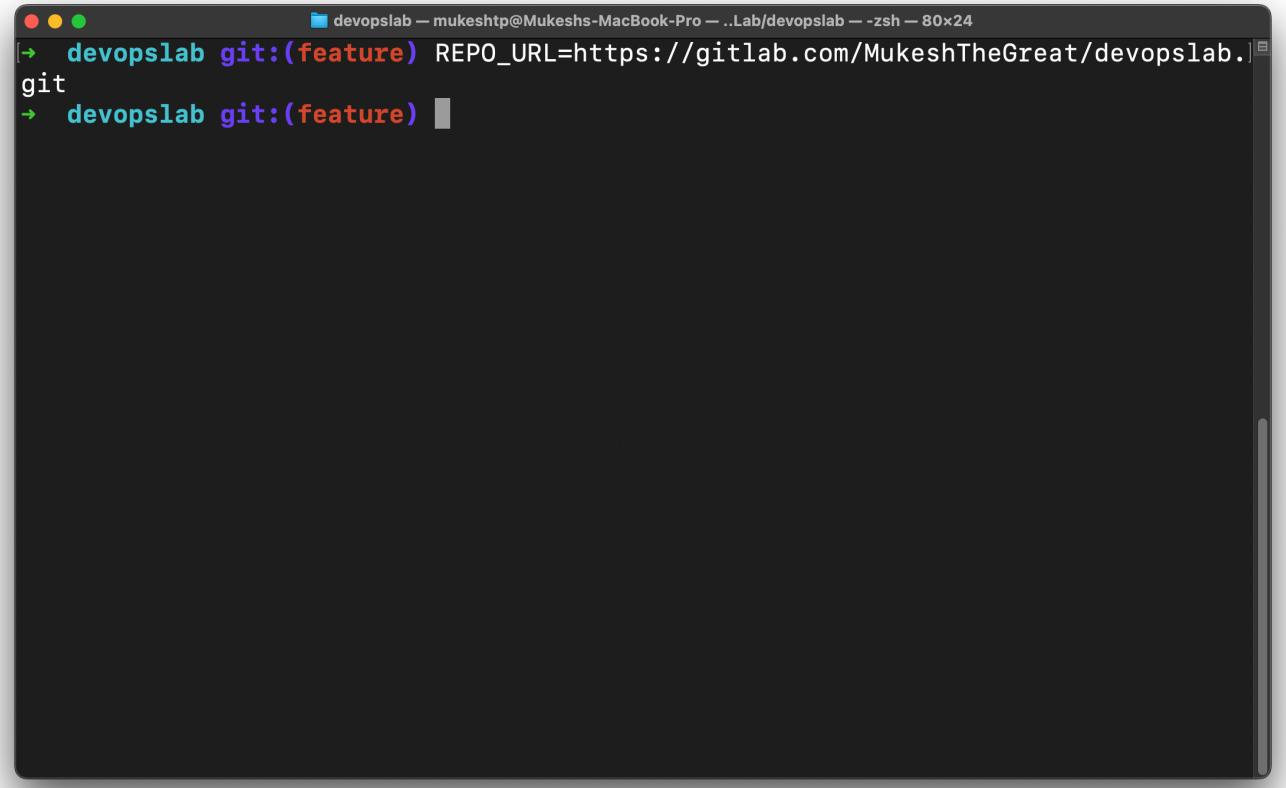
```
git checkout feature
```



4. Pushing Changes to GitLab

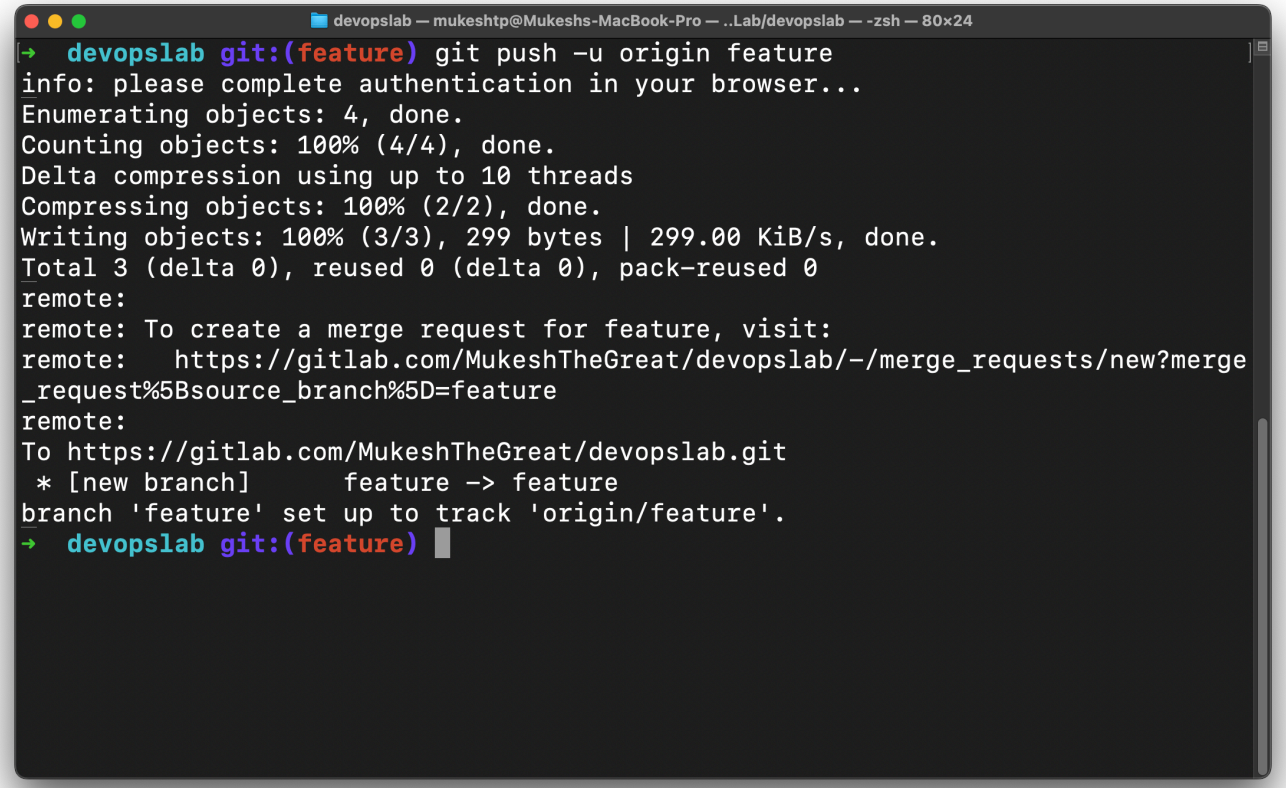
1. Add the repository URL in a variable:

```
REPO_URL=<repository_url>
```

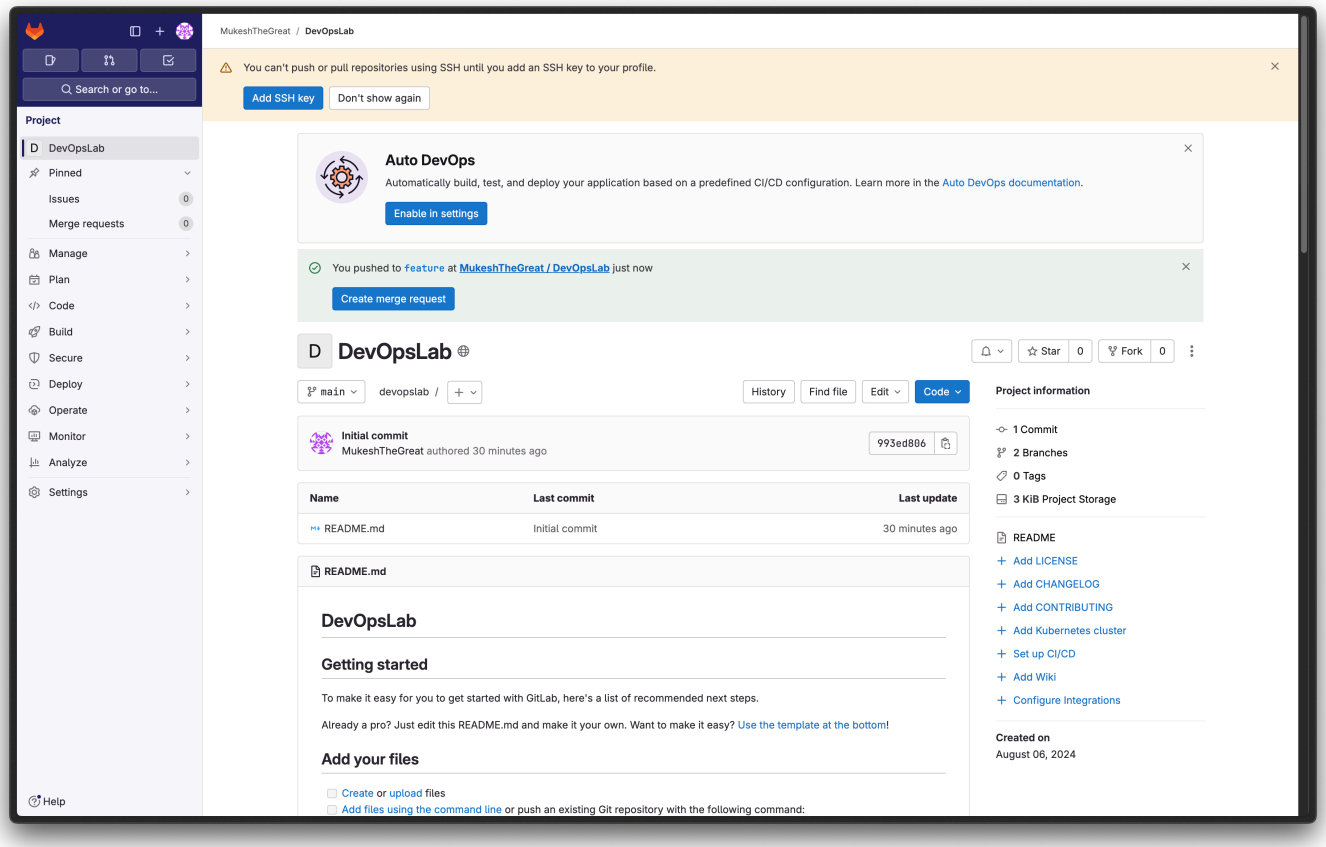


2. Push the `feature` branch to GitLab:

```
git push -u origin feature
```

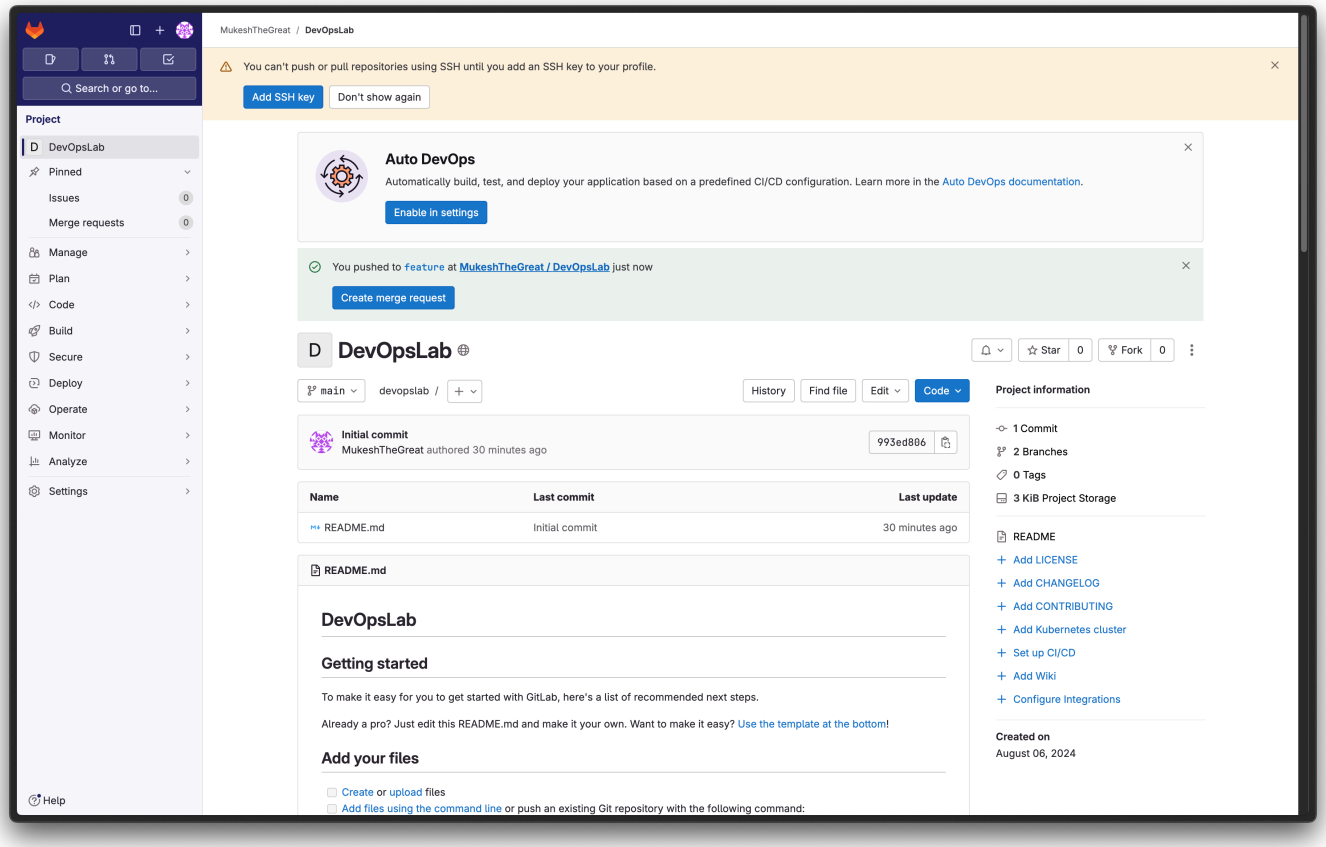


3. Check your GitLab repository to confirm that the new branch `feature` is available.

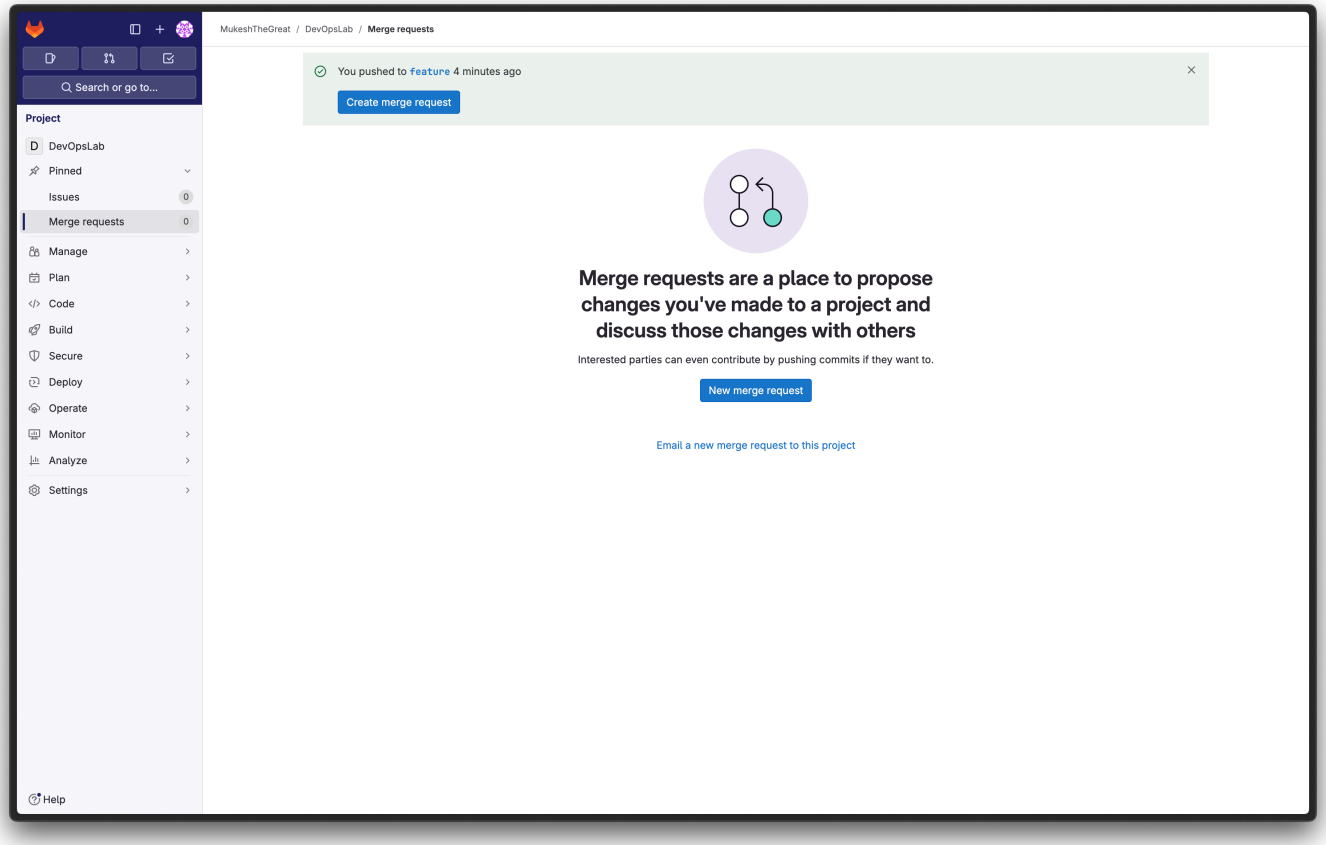


5. Collaborating through Merge Requests

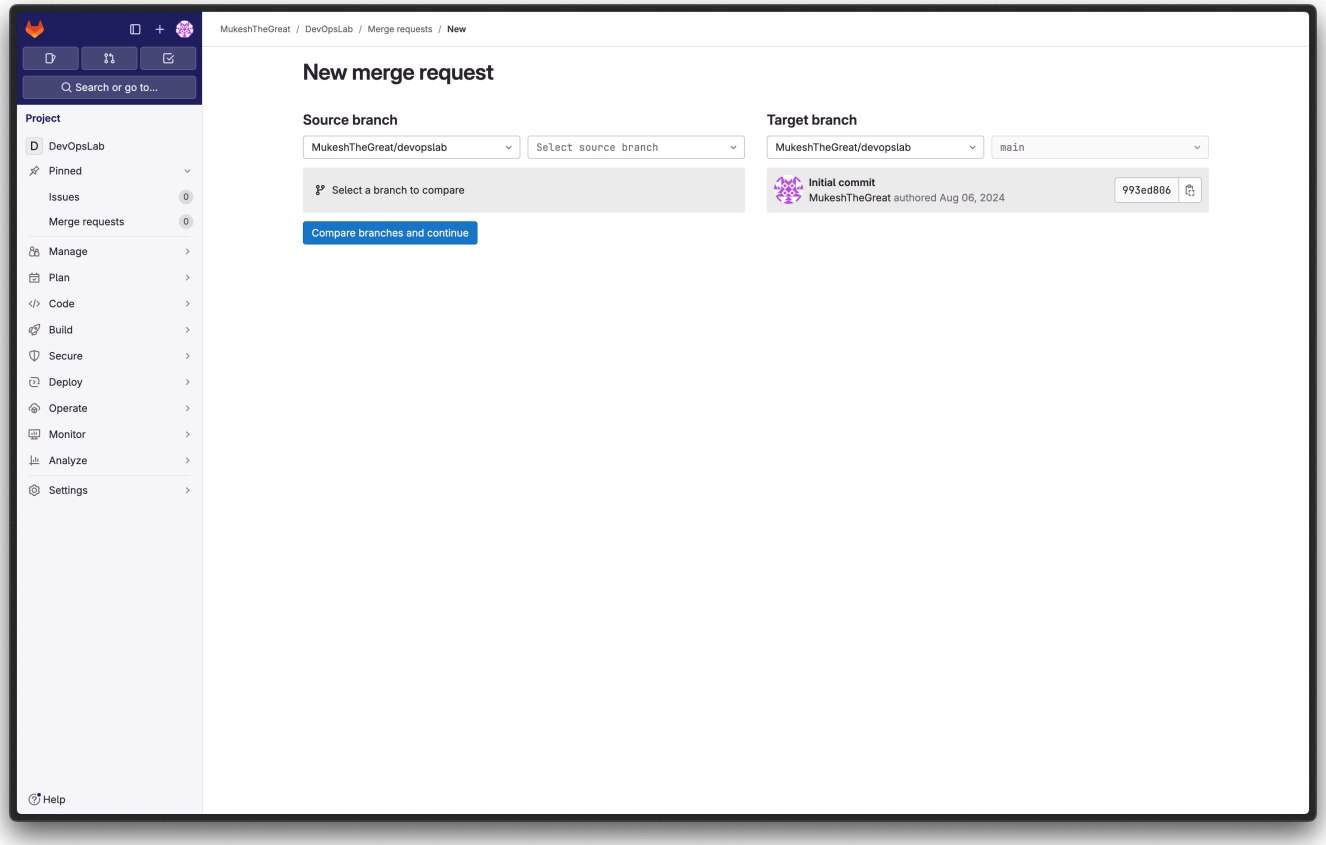
1. Go to your GitLab repository.



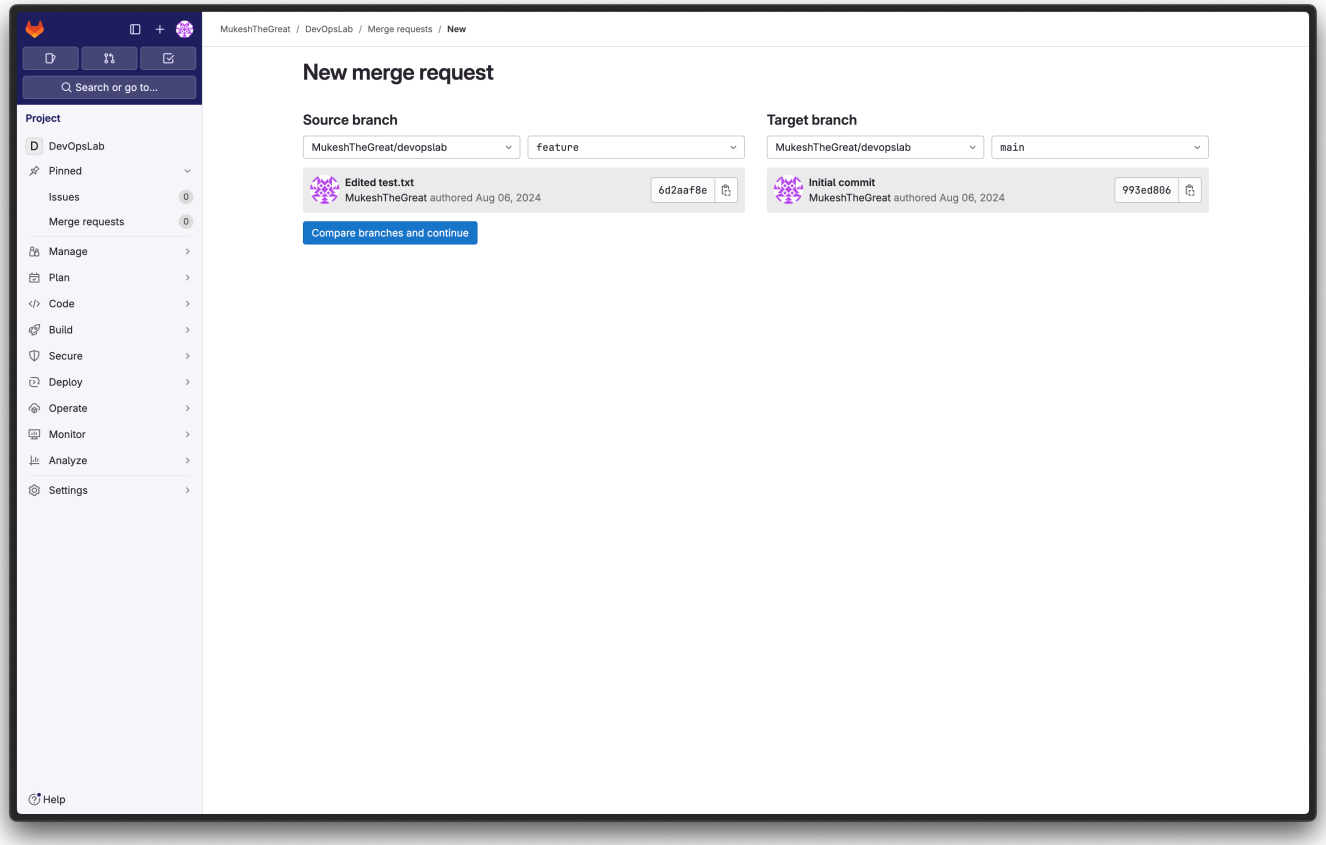
2. Click on the `Merge Requests` tab.



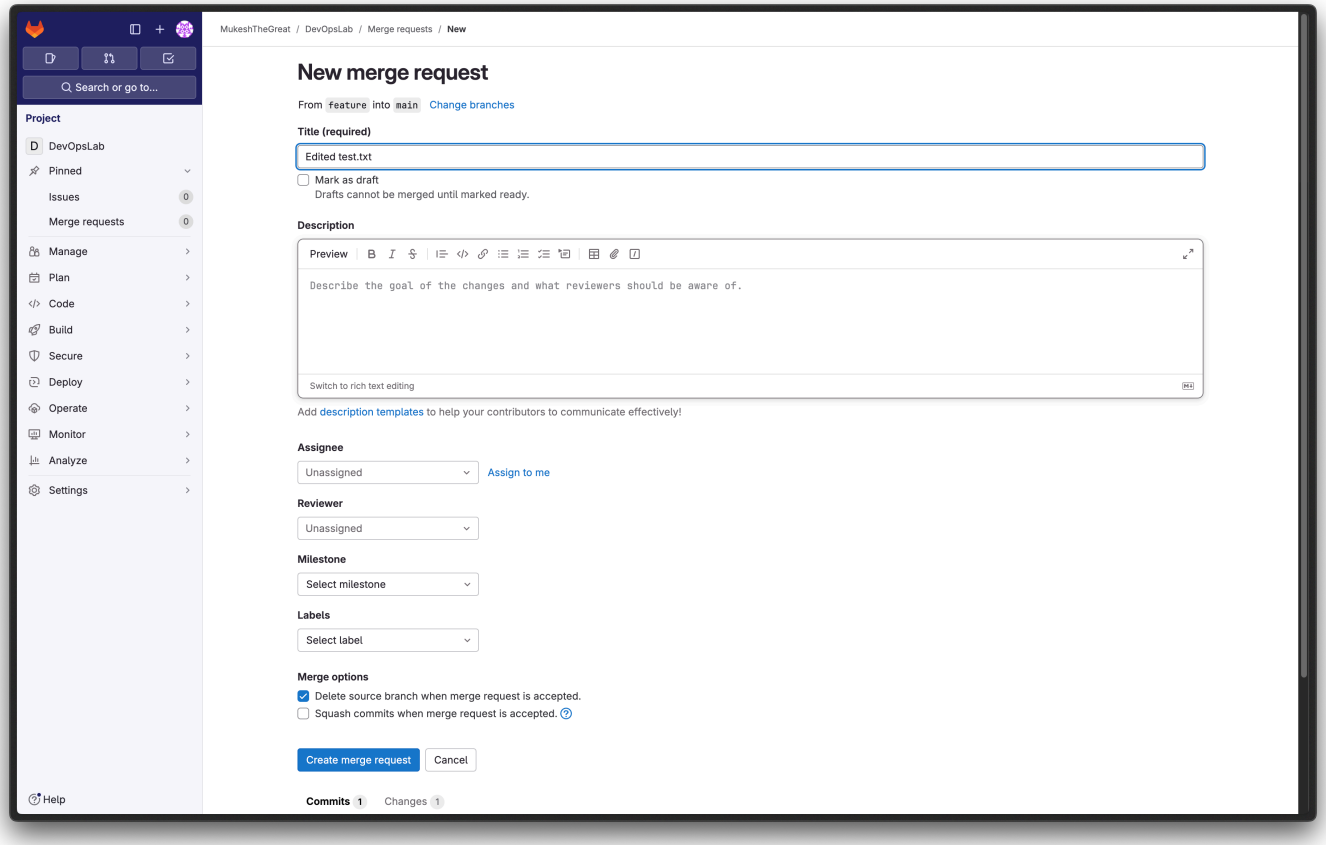
3. Click on the `New merge request` button.



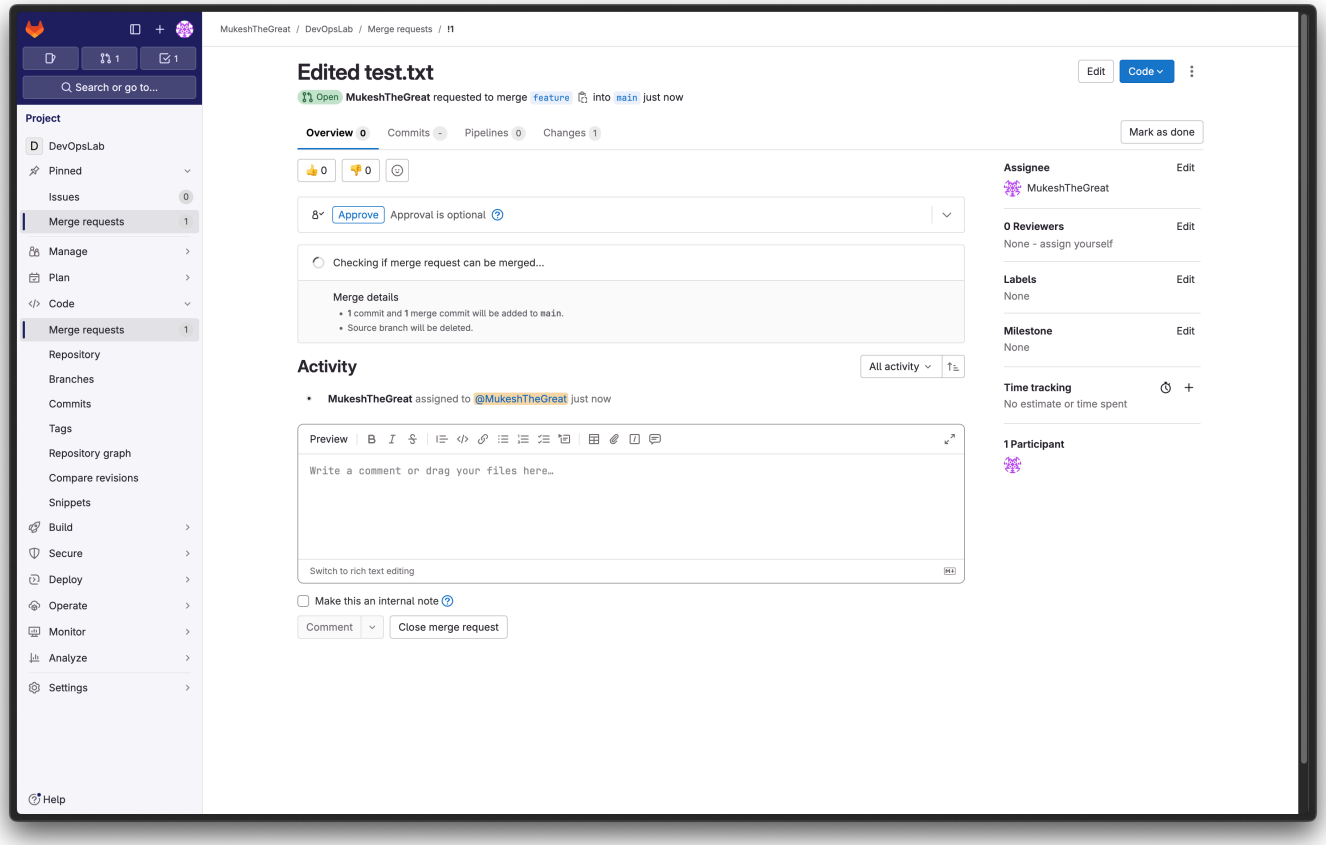
4. Select the `source branch` as `feature` and the `target branch` as `main` (or `master`).



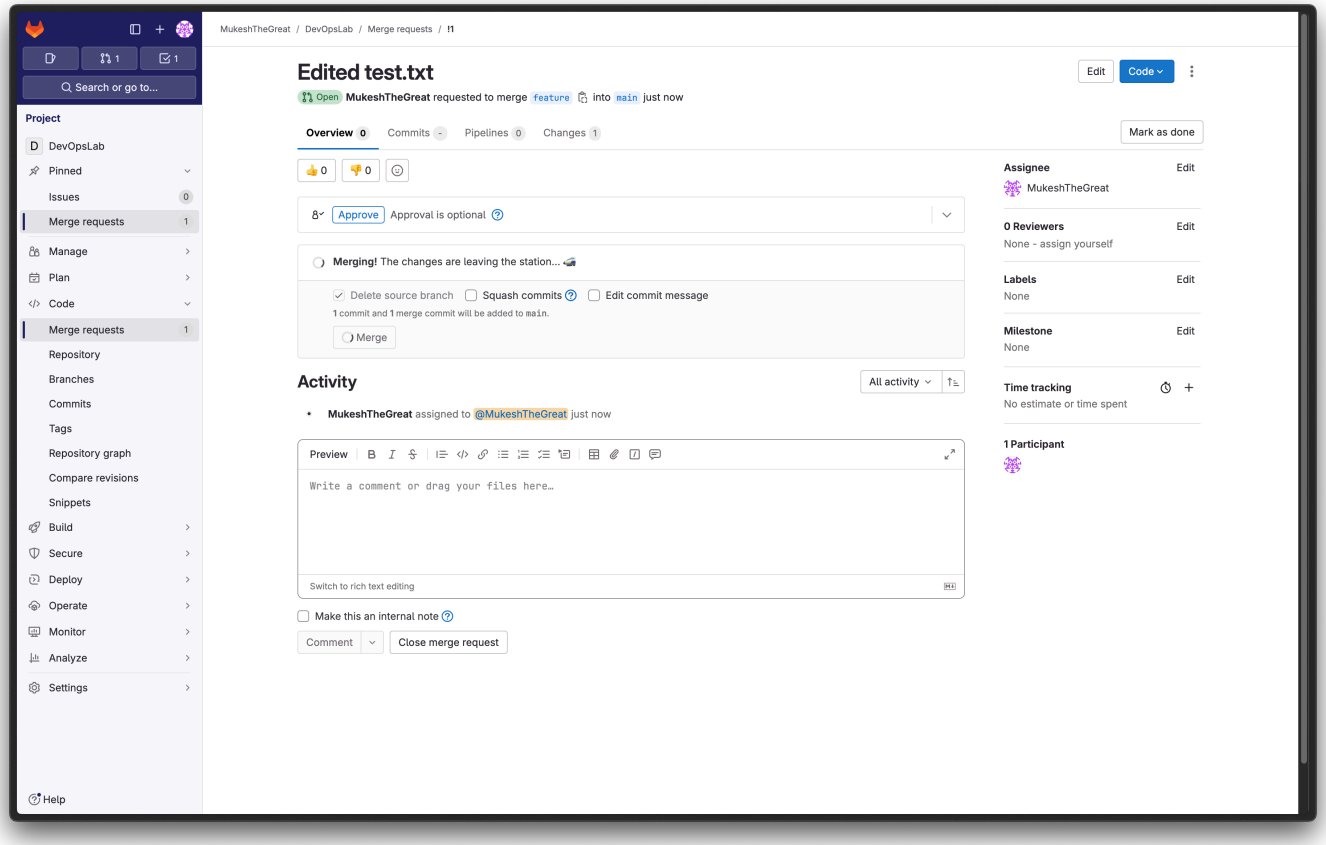
5. Click on the `Compare branches and continue` button.

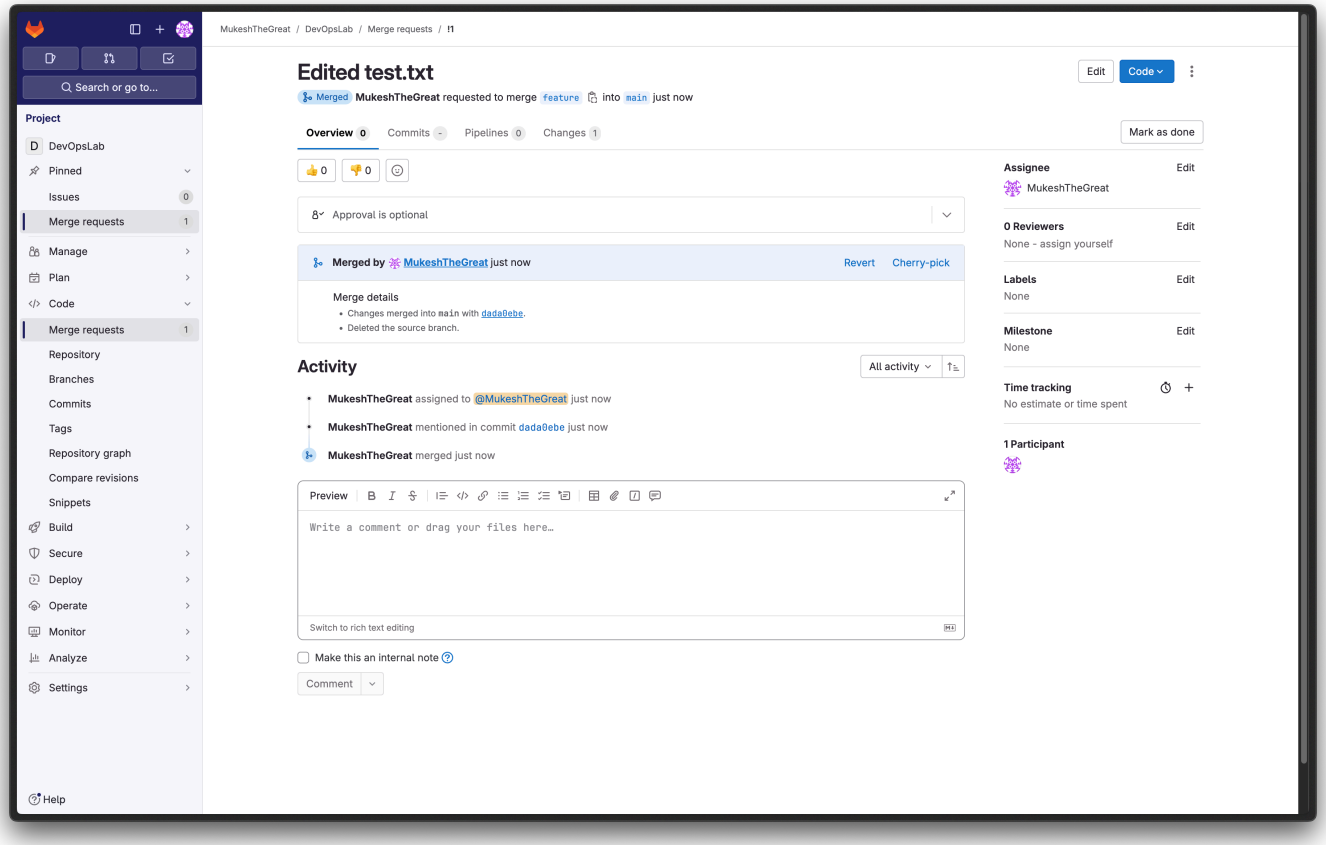


6. Fill in the details of the merge request and click on the `Create merge request` button.



7. Review the merge request and click on the `Merge` button to merge it.

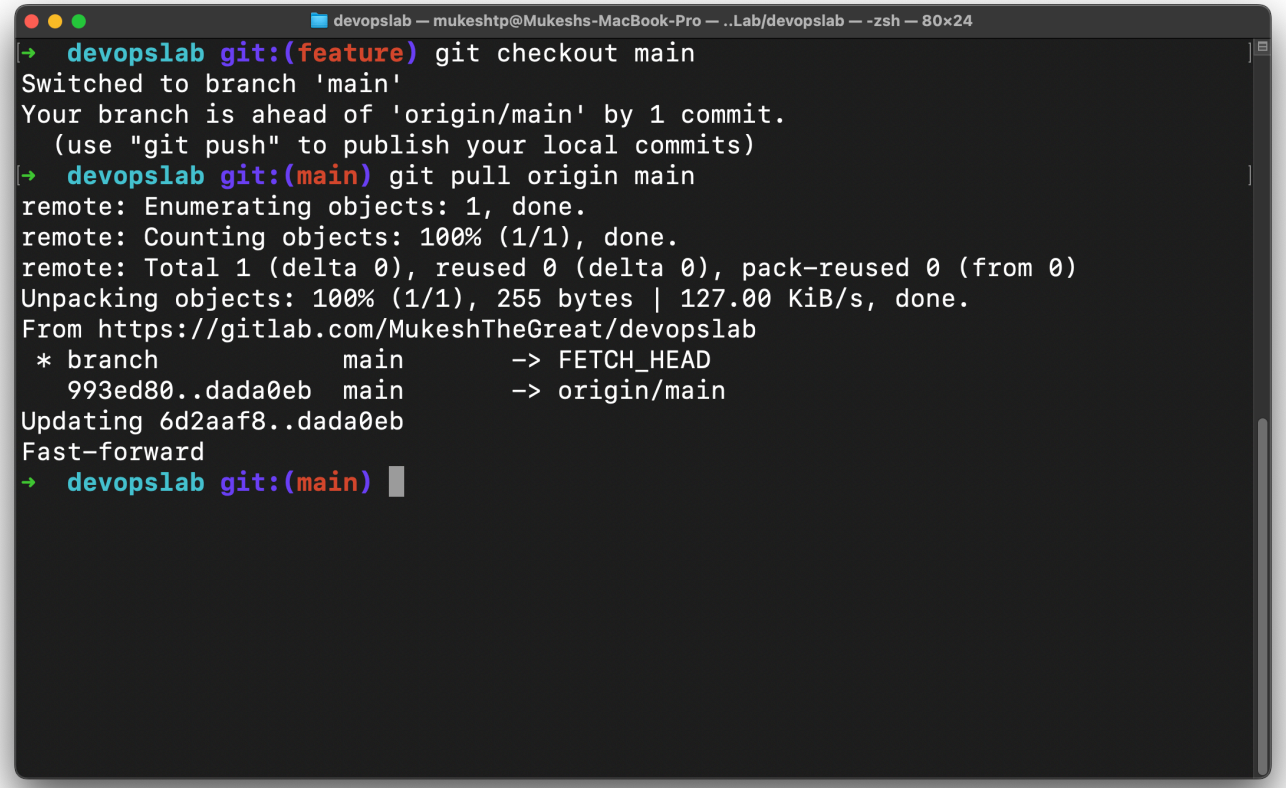




6. Syncing Changes

1. After the merge request is merged, update your local repository:

```
git checkout main
git pull origin main
```


devopslab — mukeshtp@Mukeshs-MacBook-Pro — ..Lab/devopslab — -zsh — 80×24
➜ devopslab git:(feature) git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 1 commit.
(use "git push" to publish your local commits)
➜ devopslab git:(main) git pull origin main
remote: Enumerating objects: 1, done.
remote: Counting objects: 100% (1/1), done.
remote: Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (1/1), 255 bytes | 127.00 KiB/s, done.
From https://gitlab.com/MukeshTheGreat/devopslab
* branch main -> FETCH_HEAD
993ed80..dada0eb main -> origin/main
Updating 6d2aaf8..dada0eb
Fast-forward
➜ devopslab git:(main)