

Міністерство освіти і науки України
Західноукраїнський національний університет
Факультет комп'ютерних інформаційних технологій

Кафедра ІОСУ

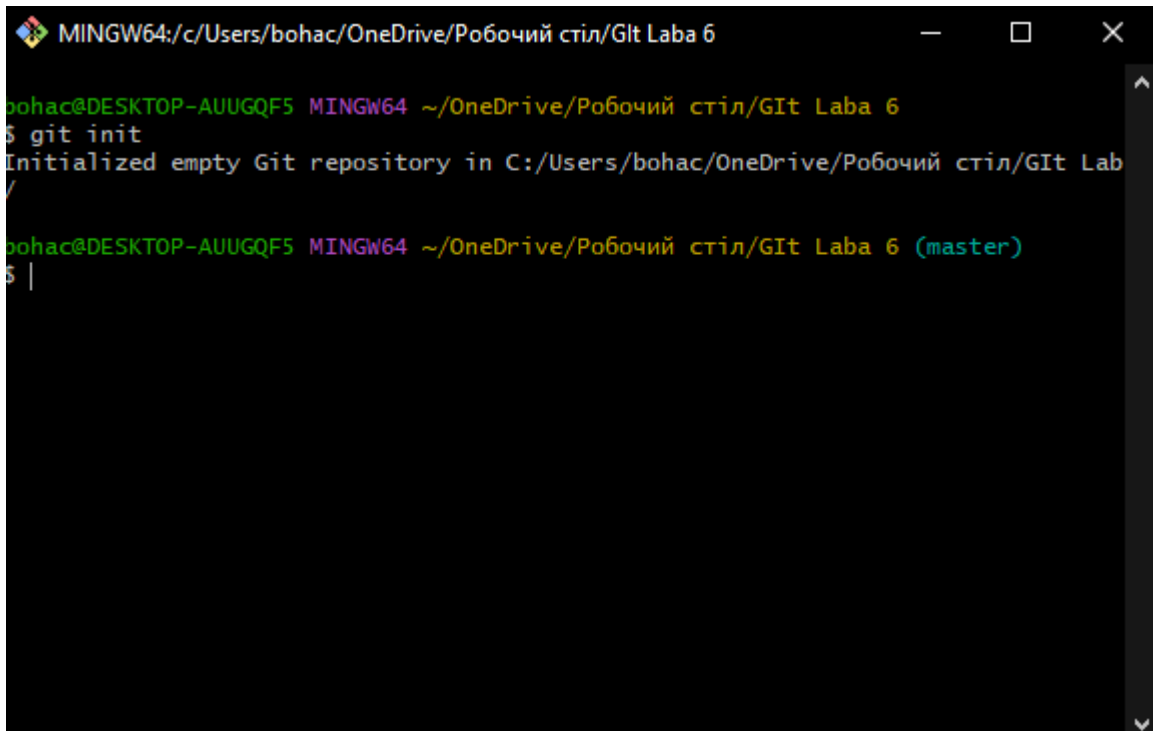
Лабораторна робота №6.1
з дисципліни «Основи комп'ютерних наук»

Виконав:
Студент групи КН-11
Богач.М.С.

Тернопіль 2023

Хід роботи

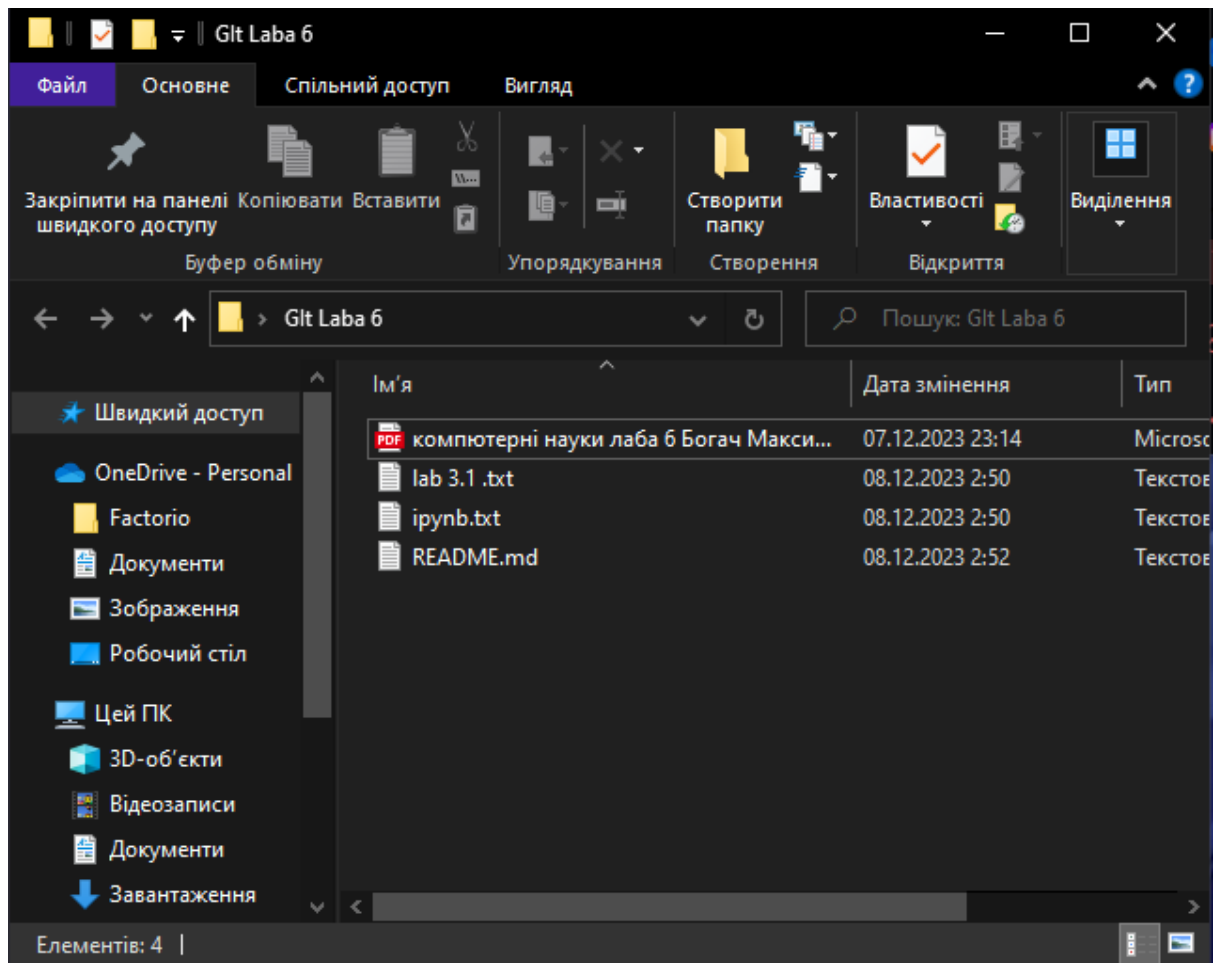
Частина 1: 1. Ініціалізація репозиторію: • Створюю нову папку для репозиторію та ініціалізую її як GIT репозиторій.(рисунок: 1)



```
MINGW64:/c/Users/bohac/OneDrive/Робочий стіл/Git Laba 6
bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6
$ git init
Initialized empty Git repository in C:/Users/bohac/OneDrive/Робочий стіл/Git Laba 6/
bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (master)
$ |
```

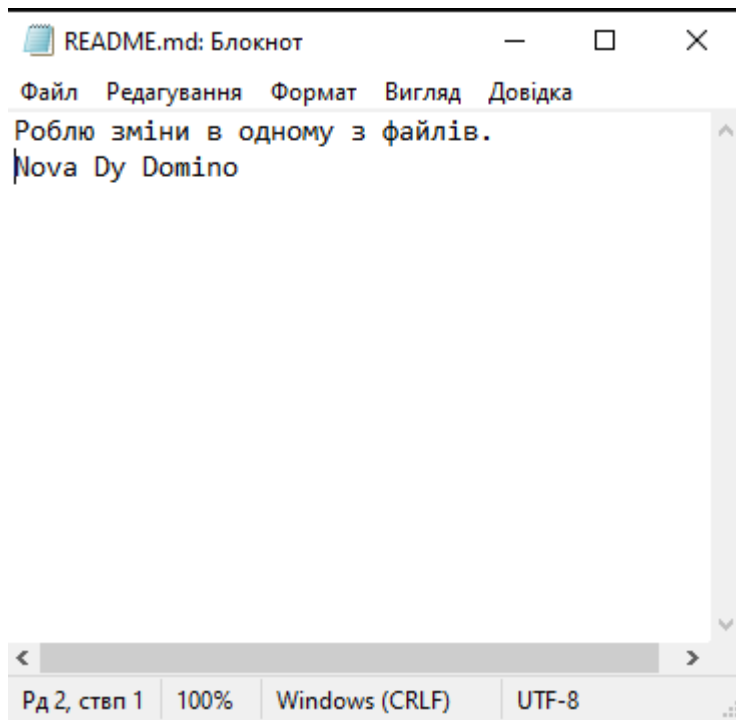
Ініціалізую git репозиторій (рисунок:1)

2. Створення та редагування файлів: • Додаю кілька файлів до репозиторію (рисунок: 2)(README.md, Лаб3.1. ірynb та звіт по ЛБ6).



Додаю декілька файлів до репозиторію (рисунок: 2)

Роблю зміни в одному з файлів.(рисунок: 3)



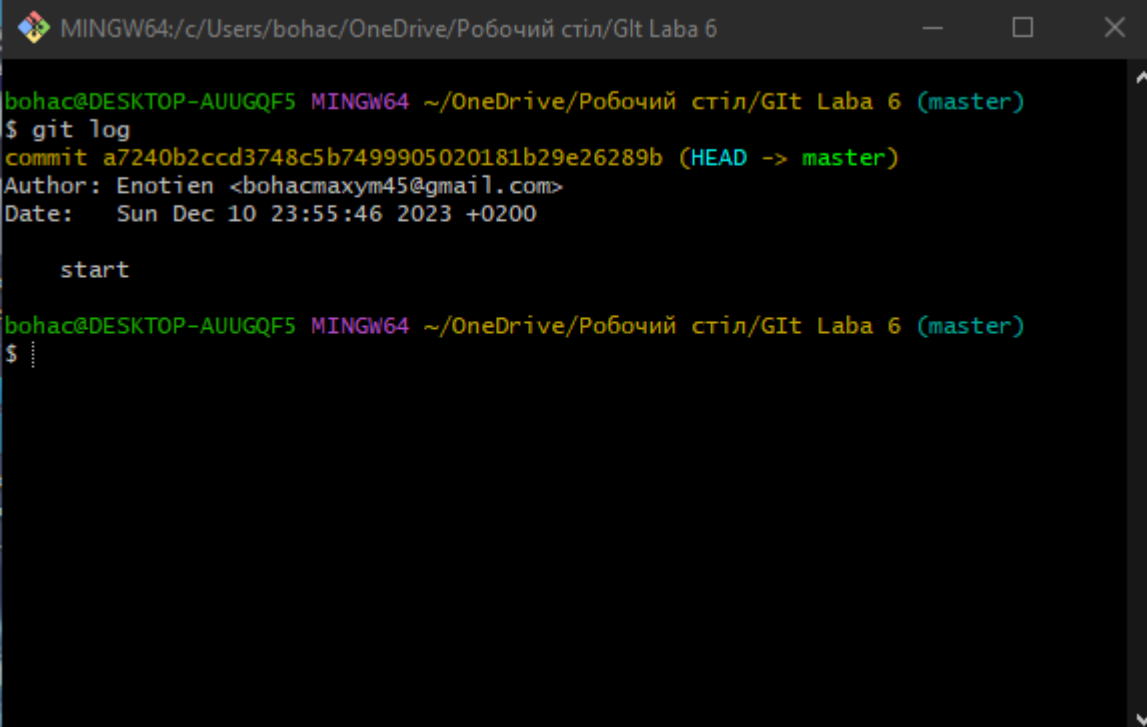
Зміни в одному із файлів(рисунок: 3)

3. Робота з комітами: • Використовую команди git add, git commit для збереження змін.(рисунок: 4)

```
MINGW64:/c:/Users/bohac/OneDrive/Робочий стіл/Git Laba 6
bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (master)
$ git add .
bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (master)
$ git commit -m "start"
[master (root-commit) a7240b2] start
4 files changed, 2 insertions(+)
create mode 100644 README.md.txt
create mode 100644 ipynb.txt.txt
create mode 100644 lab 3.1 .txt.txt
create mode 100644 "\320\272\320\276\320\274\320\277\321\216\321\202\320\265\32
1\200\320\275\321\226 \320\275\320\260\321\203\320\272\320\270 \320\273\320\260\
320\261\320\260 6 \320\221\320\276\320\263\320\260\321\207 \320\234\320\260\320\
272\321\201\320\270\320\274 \320\232\320\235-11.pdf"
bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (master)
$ |
```

Використовую git add, git commit для збереження змін.(рисунок: 4)

Переглядаю історію комітів за допомогою git log.(рисунок: 5)

A screenshot of a Windows terminal window titled "MINGW64: c:/Users/bohac/OneDrive/Робочий стіл/Git Laba 6". The terminal shows the execution of the 'git log' command. The output displays a single commit with the hash 'a7240b2ccd3748c5b7499905020181b29e26289b', which is the HEAD of the 'master' branch. The author is 'Enotien' with the email 'bohacmaxym45@gmail.com'. The commit date is 'Sun Dec 10 23:55:46 2023 +0200'. The commit message is 'start'. The prompt indicates the user is in the 'master' branch of the 'Git Laba 6' repository.

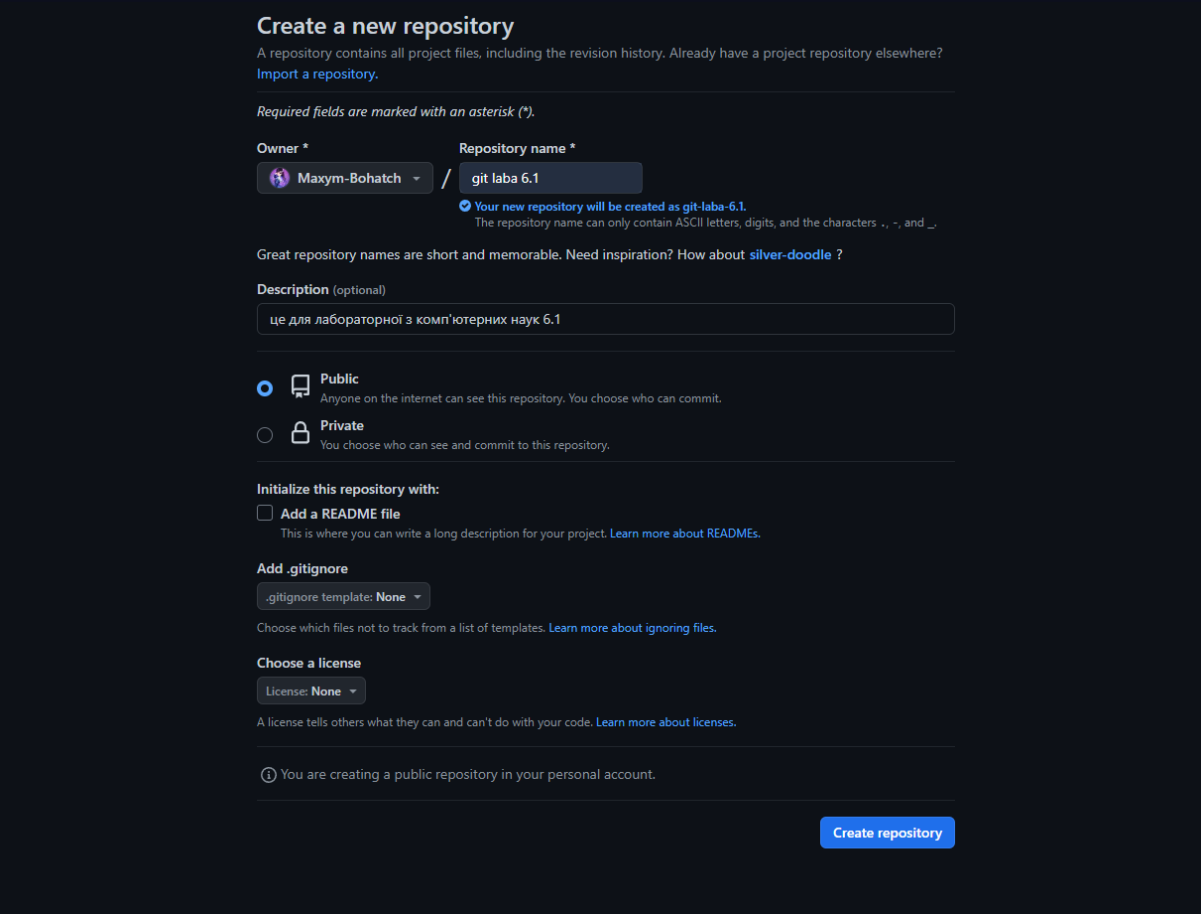
```
bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (master)
$ git log
commit a7240b2ccd3748c5b7499905020181b29e26289b (HEAD -> master)
Author: Enotien <bohacmaxym45@gmail.com>
Date:   Sun Dec 10 23:55:46 2023 +0200

    start

bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (master)
$
```

Переглядаю історію комітів за допомогою git log (рисунок: 5)

Частина 2: Робота з GitHub 1. Створення репозиторію на GitHub: Створюю новий репозиторій на GitHub.(рисунок: 6)



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk ().*

Owner * / Repository name *

Maxym-Bohatch / git laba 6.1

✓ Your new repository will be created as git-laba-6.1.
The repository name can only contain ASCII letters, digits, and the characters -, ., and _.

Great repository names are short and memorable. Need inspiration? How about [silver-doodle](#) ?

Description (optional)

це для лабораторної з комп'ютерних наук 6.1

☒ Public
Anyone on the internet can see this repository. You choose who can commit.

☐ Private
You choose who can see and commit to this repository.

Initialize this repository with:

☒ Add a README file
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None

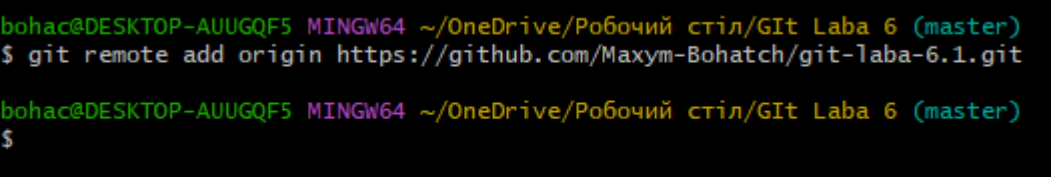
A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

❗ You are creating a public repository in your personal account.

Create repository

Створюю новий репозиторій на GitHub.(рисунок: 6)

Зв'язую локальний репозиторій з віддаленим на GitHub за допомогою `git remote add origin [URL]`.(рисунок: 7)

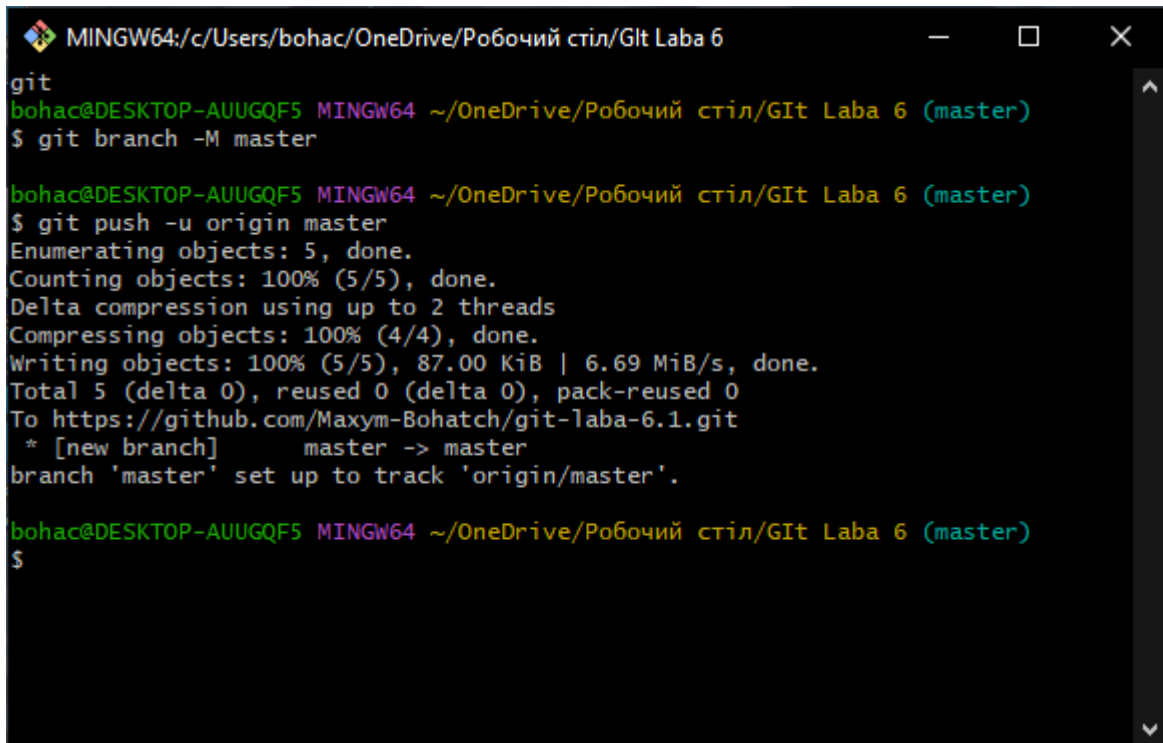


```
bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (master)
$ git remote add origin https://github.com/Maxym-Bohatch/git-laba-6.1.git

bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (master)
$
```

Зв'язую локальний репозиторій з віддаленим на GitHub за допомогою `git remote add origin [URL]`.(рисунок: 7)

2. Push та Pull: • Зробіть git push для відправлення локальних змін на GitHub.(рисунок:8,8,11)

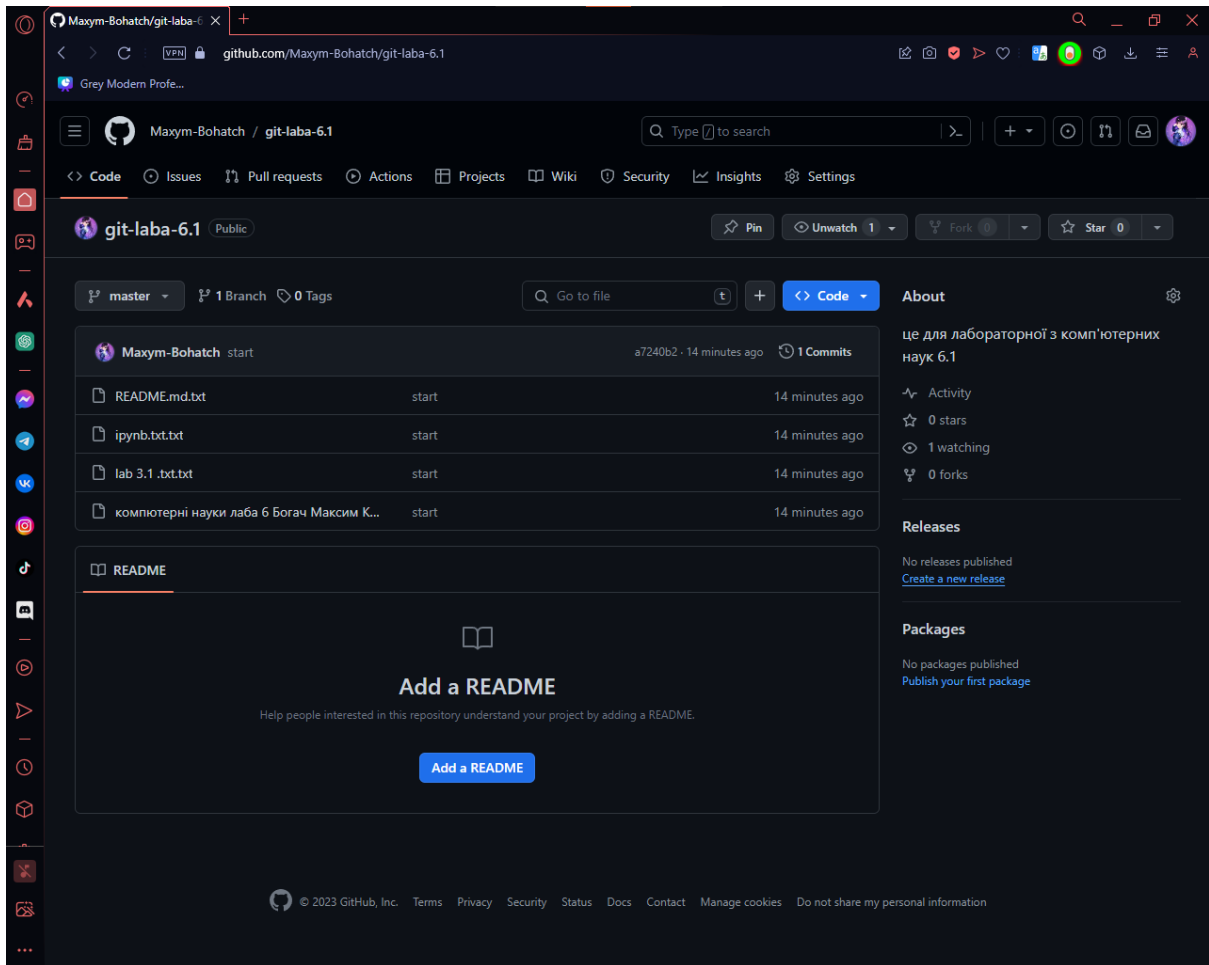
A screenshot of a Windows command prompt window titled "MINGW64: c:/Users/bohac/OneDrive/Робочий стіл/Git Laba 6". The prompt shows the user "bohac@DESKTOP-AUUGQF5" in the "MINGW64" environment, located at "~ /OneDrive/Робочий стіл/Git Laba 6 (master)". The user enters the command "\$ git branch -M master". The prompt then shows the user entering "\$ git push -u origin master". The output of the command is displayed: "Enumerating objects: 5, done.", "Counting objects: 100% (5/5), done.", "Delta compression using up to 2 threads", "Compressing objects: 100% (4/4), done.", "Writing objects: 100% (5/5), 87.00 KiB | 6.69 MiB/s, done.", "Total 5 (delta 0), reused 0 (delta 0), pack-reused 0", "To https://github.com/Maxym-Bohatch/git-laba-6.1.git", "* [new branch] master -> master", and "branch 'master' set up to track 'origin/master'.". The prompt then shows the user entering "\$" again.

```
git
bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (master)
$ git branch -M master

bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (master)
$ git push -u origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 2 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 87.00 KiB | 6.69 MiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Maxym-Bohatch/git-laba-6.1.git
* [new branch] master -> master
branch 'master' set up to track 'origin/master'.

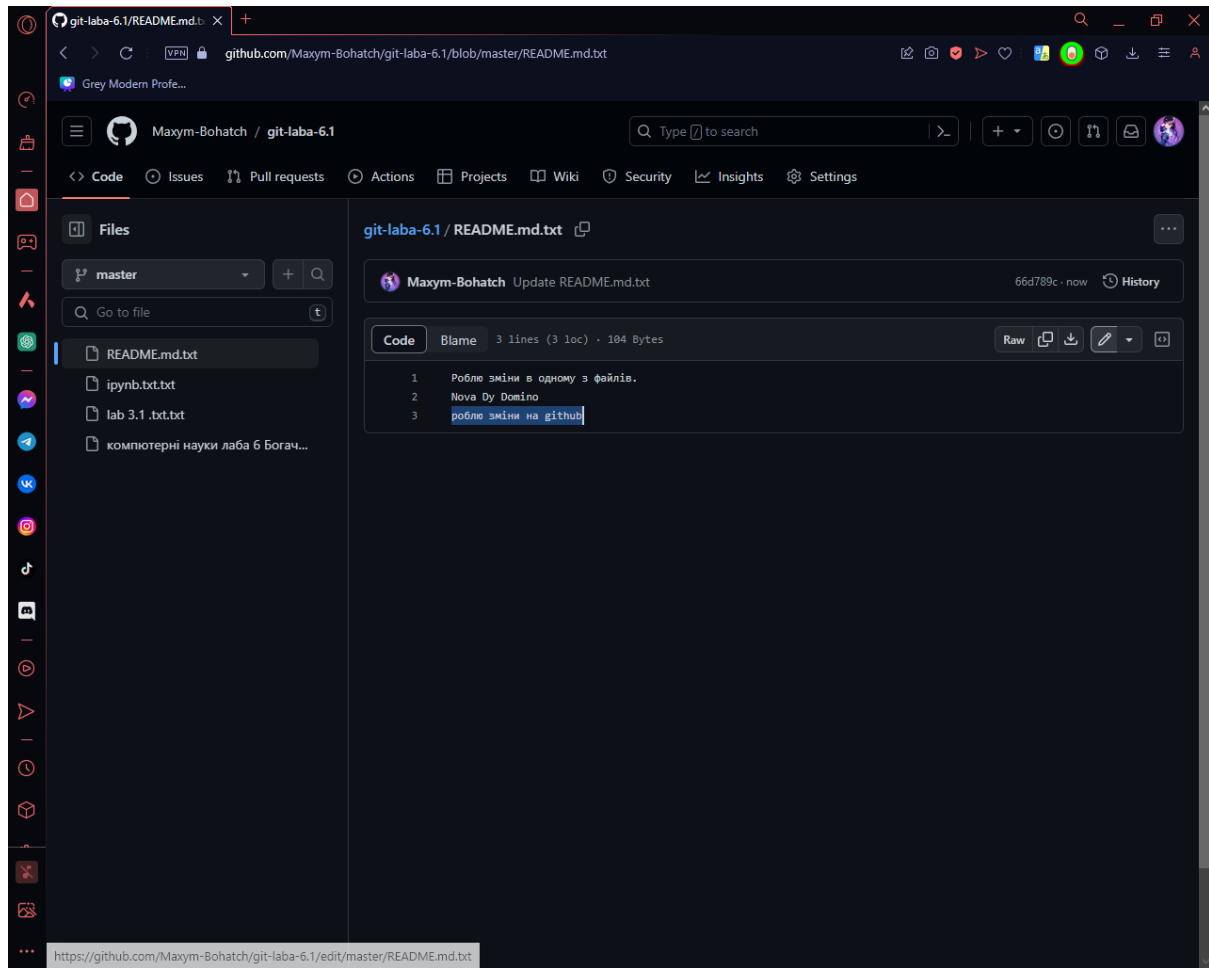
bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (master)
$
```

Використовую git push для відправлення локальних змінних на GitHub(рисунок:8)



Перевіряю чи файли дійсно завантажені(рисунок: 9)

- Внесіть зміни в файл на GitHub і виконайте `git pull`, щоб синхронізувати зміни з локальним репозиторієм. (рисунк:10,11,12)



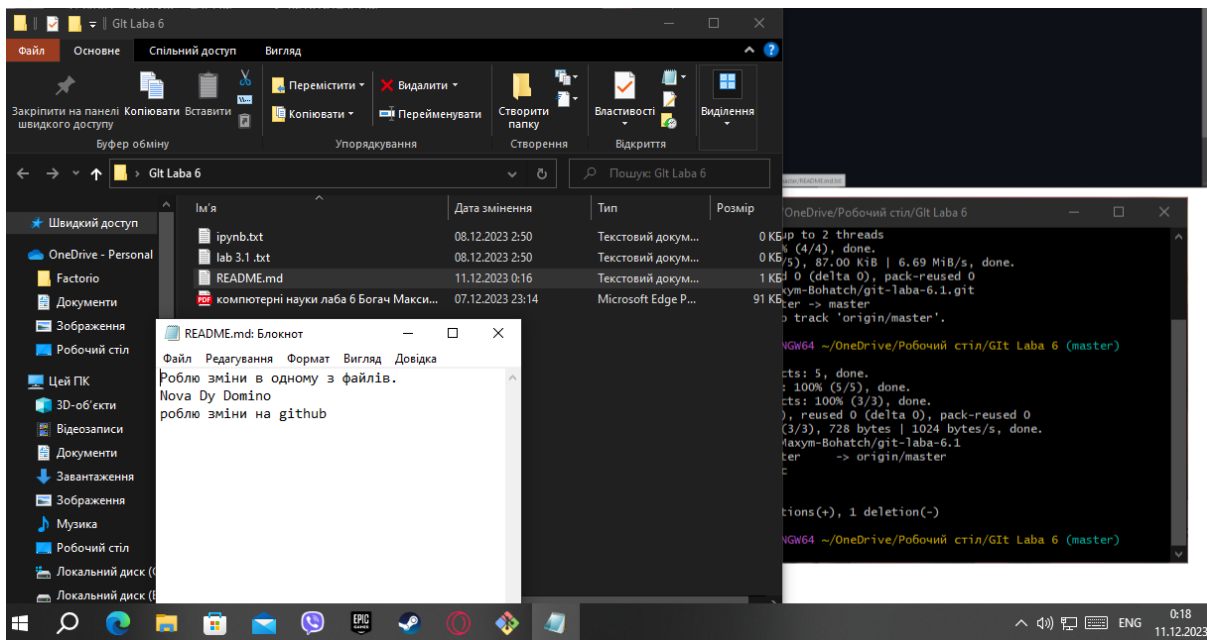
Вношу зміни в файл через GitHub(з права кнопка для редагування)(рисунк: 10)

```
MINGW64: c:/Users/bohac/OneDrive/Робочий стіл/Git Laba 6
Delta compression using up to 2 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 87.00 KiB | 6.69 MiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Maxym-Bohatch/git-laba-6.1.git
* [new branch]      master -> master
branch 'master' set up to track 'origin/master'.

bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (master)
$ git pull
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 728 bytes | 1024 bytes/s, done.
From https://github.com/Maxym-Bohatch/git-laba-6.1
   a7240b2..66d789c  master    -> origin/master
Updating a7240b2..66d789c
Fast-forward
 README.md.txt | 3 ++-
 1 file changed, 2 insertions(+), 1 deletion(-)

bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (master)
$
```

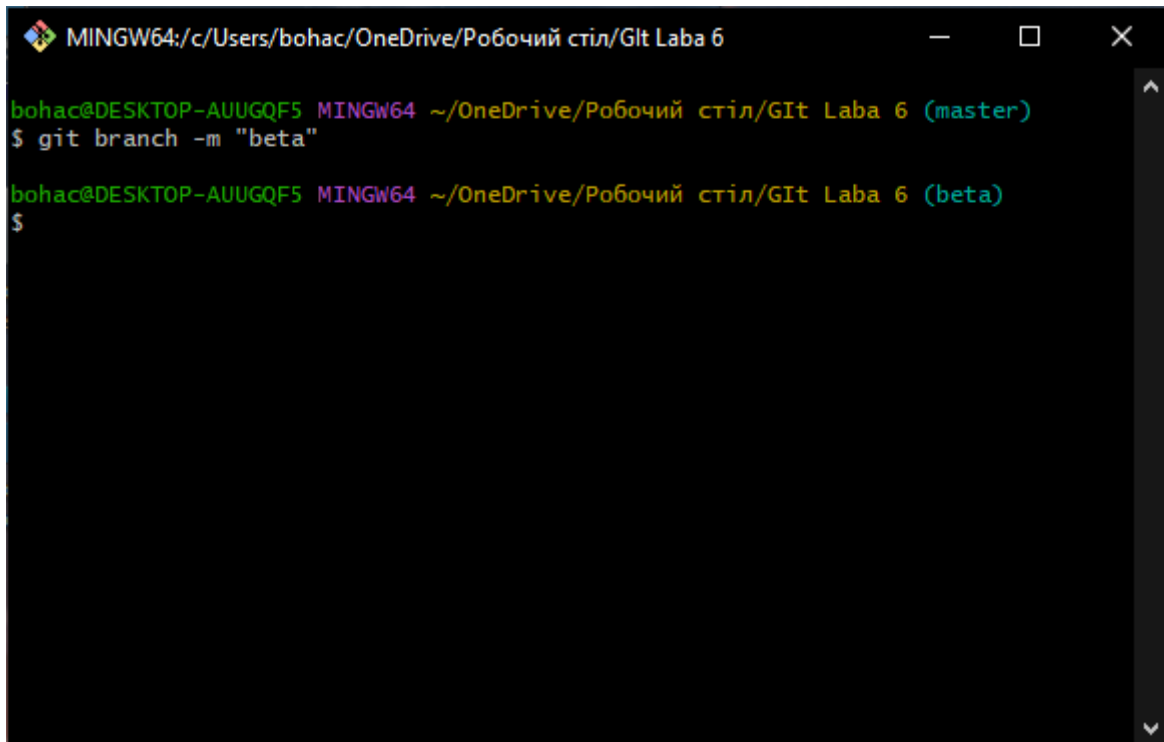
Використовую git pull для синхронізації файлів(рисунок: 11)



Перевіряю наявність зміни в файлах(рисунок: 12)

Частина 3: Співпраця та Гілки

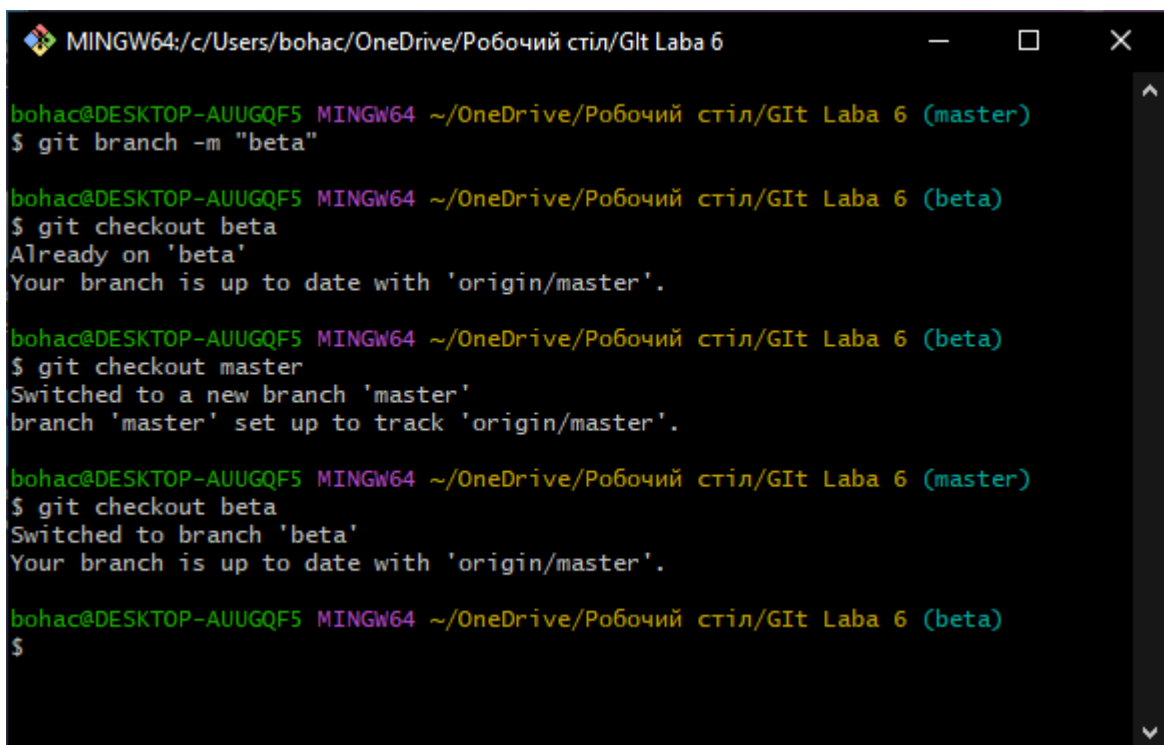
1. Робота з гілками: • Створюю нову гілку за допомогою git branch [назва гілки] та переключаюся на неї за допомогою git checkout [назва гілки].(рисунок: 13,14)



```
MINGW64:/c/Users/bohac/OneDrive/Робочий стіл/Git Laba 6
bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (master)
$ git branch -m "beta"

bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (beta)
$
```

Створюю нову гілку за допомогою `git branch [назва гілки]` .(рисунок: 13)



```
MINGW64:/c/Users/bohac/OneDrive/Робочий стіл/Git Laba 6
bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (master)
$ git branch -m "beta"

bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (beta)
$ git checkout beta
Already on 'beta'
Your branch is up to date with 'origin/master'.

bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (beta)
$ git checkout master
Switched to a new branch 'master'
branch 'master' set up to track 'origin/master'.

bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (master)
$ git checkout beta
Switched to branch 'beta'
Your branch is up to date with 'origin/master'.

bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (beta)
$
```

Переключаюся на неї за допомогою `git checkout [назва гілки]` .(рисунок: 14)

Вношу зміни в гілці та роблю коміт(рисунок: 15)

```
bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (beta)
$ git add .

bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (beta)
$ git commit -m "beta"
[beta 69b2239] beta
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 "\\320\\267\\320\\274\\321\\226\\320\\275\\320\\270 \\320\\262 \\320\\263\\321\\226\\320\\273\\321\\206\\321\\226.txt"
```

Вношу зміни в гілці та роблю коміт(рисунок: 15)

2. Pull Request та Merge:

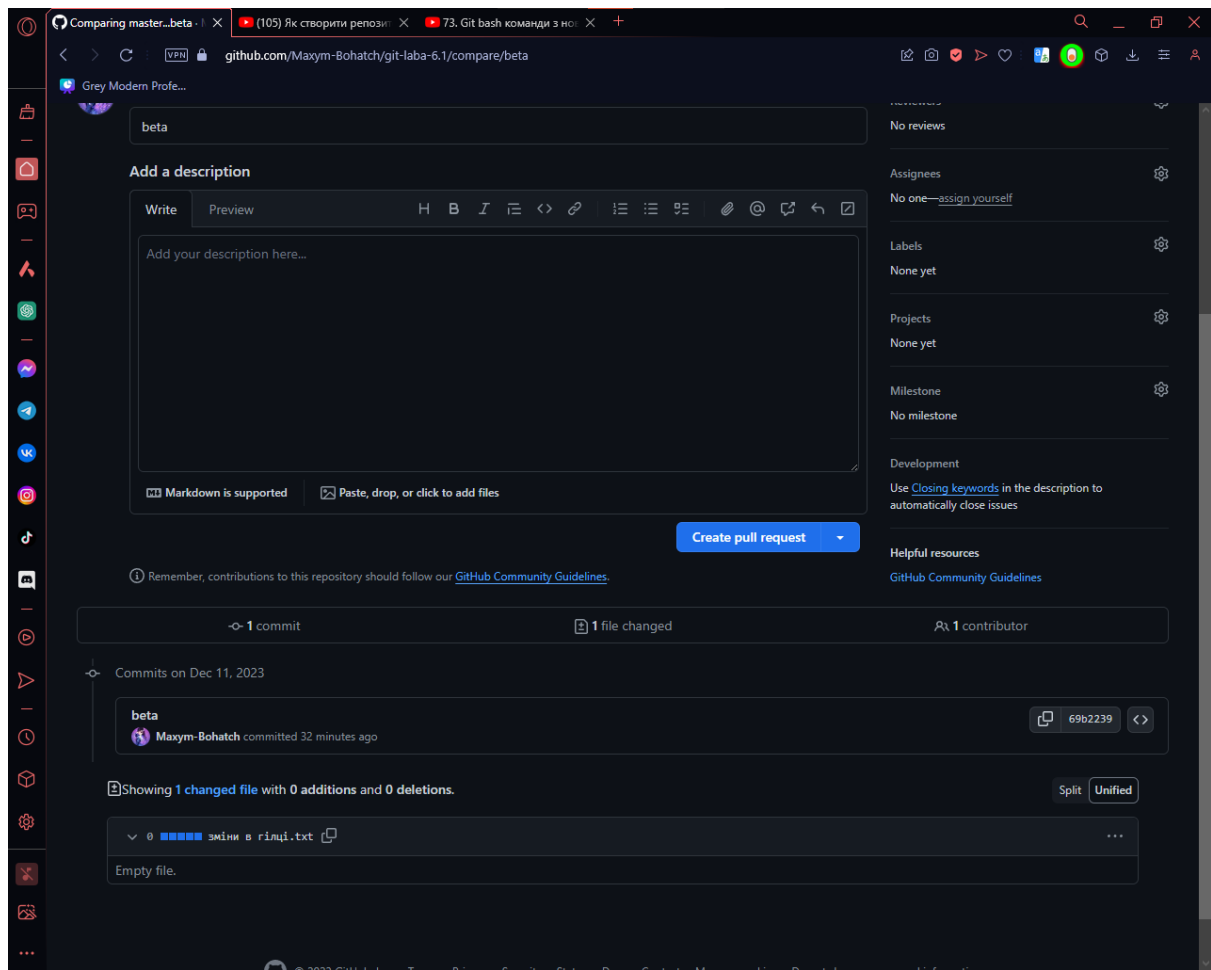
Відправляю зміни з гілки на GitHub за допомогою git push(рисунок: 16)

```
MINGW64:/c/Users/bohac/OneDrive/Робочий стіл/Git Laba 6
git push origin
bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (beta)
$ git push -u origin beta
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 248 bytes | 124.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote:
remote: Create a pull request for 'beta' on GitHub by visiting:
remote:   https://github.com/Maxym-Bohatch/git-laba-6.1/pull/new/beta
remote:
To https://github.com/Maxym-Bohatch/git-laba-6.1.git
 * [new branch]      beta -> beta
branch 'beta' set up to track 'origin/beta'.

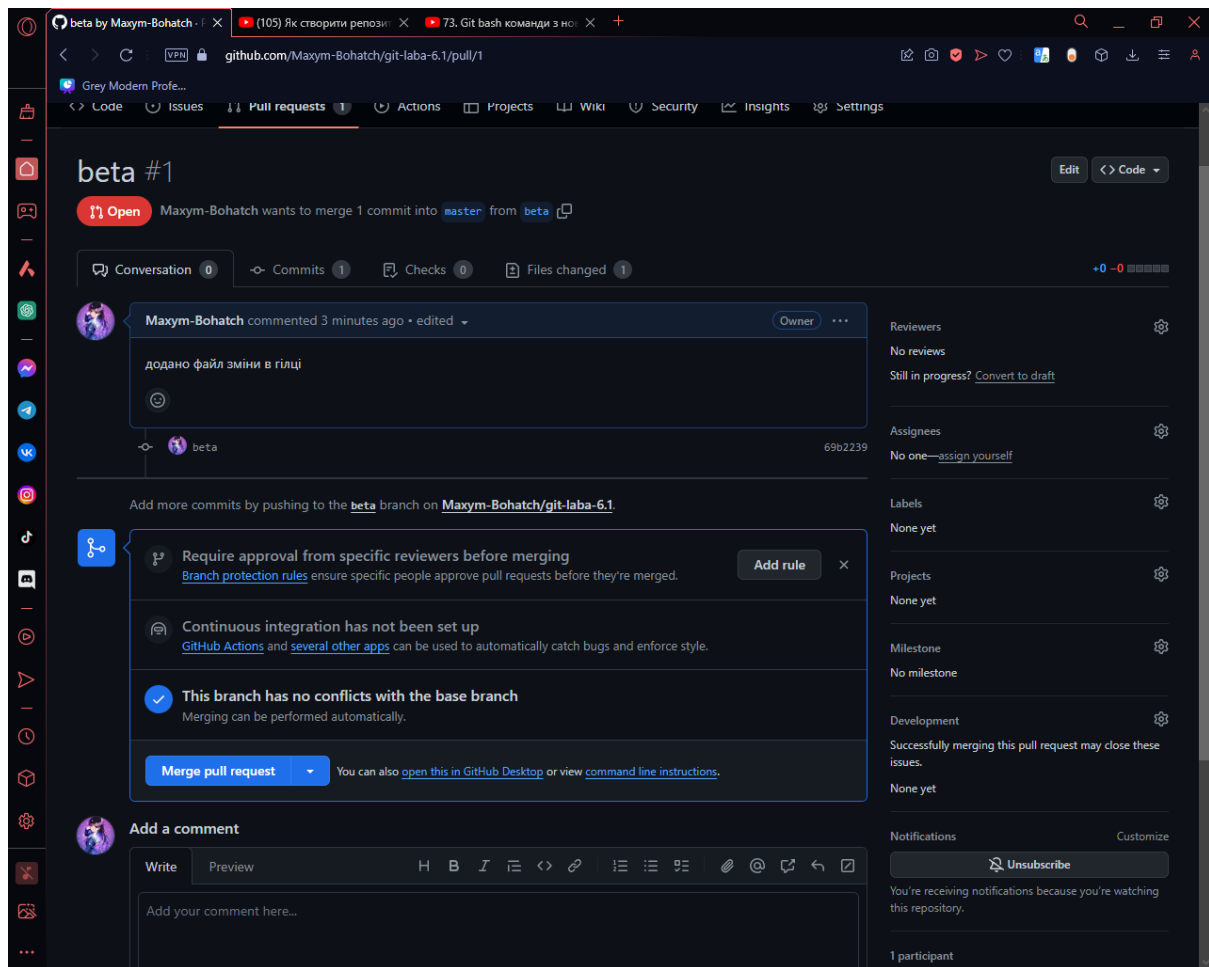
bohac@DESKTOP-AUUGQF5 MINGW64 ~/OneDrive/Робочий стіл/Git Laba 6 (beta)
$ |
```

Відправляю зміни з гілки на GitHub за допомогою git push(рисунок: 16)

На GitHub створюю Pull Request для об'єднання гілки з основною гілкою.(рисунок: 17,18)

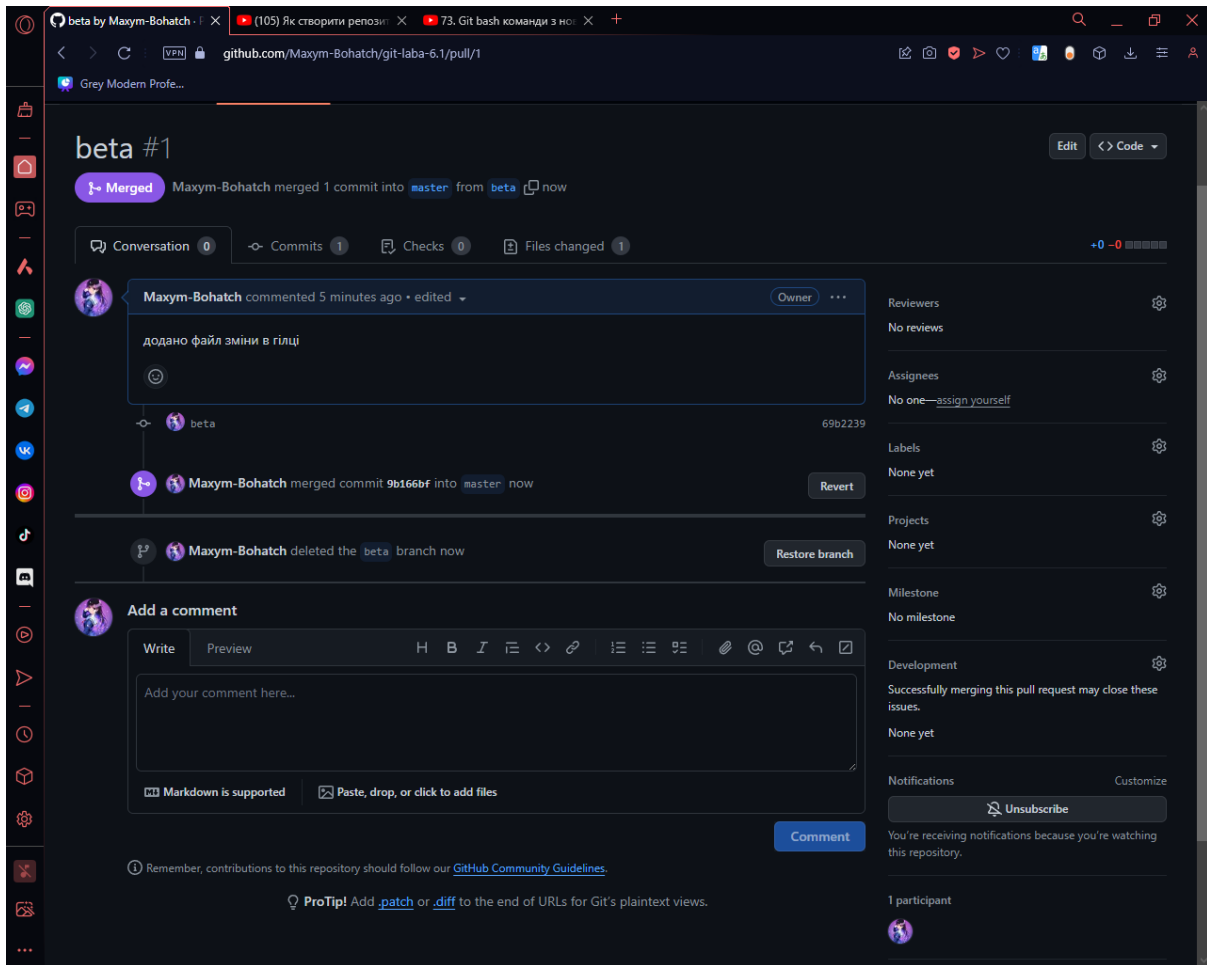


На GitHub створюю Pull Request для об'єднання гілки з основною гілкою.(рисунок: 17)

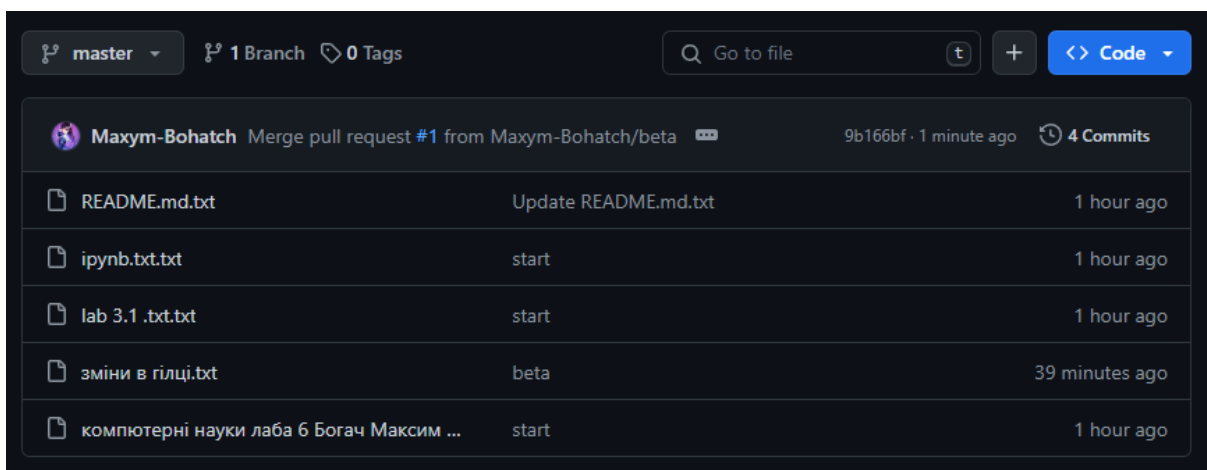


На GitHub створюю Pull Request для об'єднання гілки з основною гілкою.(рисунок: 18)

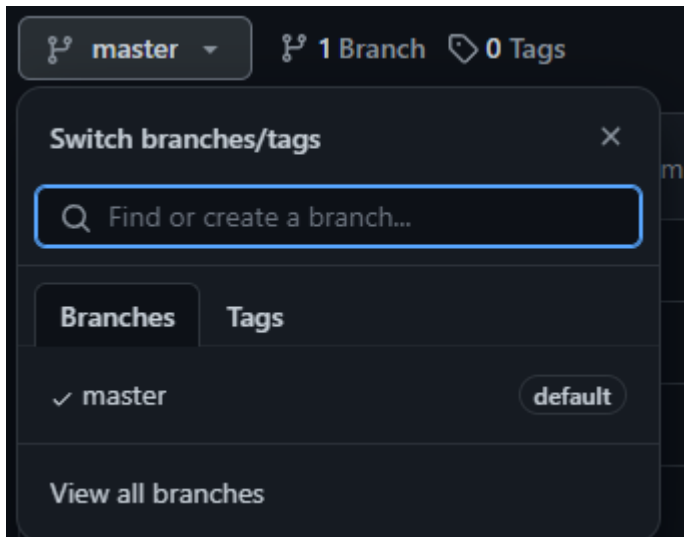
Опишіть зміни та виконайте Merge Pull Request.(рисунок:19,20,21)



виконую Merge Pull Request.(рисунок: 19)



перевіряю наявність змін у файлах у гілці master(рисунок: 20)



перевіряю чи видалилася гілка(рисунок: 21)

Звіт:

Я вважаю що систему GitHub можна використовувати для збереження програми у хмарному середовищі, на випадок несправності обладнання. Також для контролю версій у разі помилки чи випадково стирання фрагменту, або цілої програми.

Висновок: Я навчився основам роботи з GIT та GitHub, включаючи управління версіями, роботу з гілками, та створення Pull Request. А також отримав кілька цікавих безсонних ночей.