



Міністерство освіти і науки України
Національний технічний університет України “Київський
політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки Кафедра
інформаційних систем та технологій

Лабораторна робота №1

Технології розроблення програмного забезпечення

Тема: «Основи роботи з Git»

Виконав:

Студент групи ІА-31

Калиновський В.О

Перевірив:

Мягкий Михайло Юрійович

Тема: Основи роботи з Git

Мета: Ознайомитися з базовими можливостями системи контролю версій Git. Навчитися створювати новий репозиторій, ініціалізувати коміт, створювати та перемикатися між гілками, додавати й фіксувати зміни у файлах, а також виконувати злиття гілок із розв'язанням конфліктів. У процесі роботи закріпити практичні навички роботи з різними командами

Теоретичні відомості

Git — це розподілена система контролю версій, яка дозволяє відслідковувати зміни у файлах, працювати над проектами в команді та зберігати історію розробки. Вона використовується для зручного управління програмним кодом і дозволяє розробникам повертатися до попередніх версій, створювати паралельні гілки розвитку та об'єднувати результати роботи.

Основні поняття Git:

- Repository - сховище, в якому зберігається весь проект та історія його змін. Репозиторій може бути локальним (на комп'ютері користувача) та віддаленим (на сервері, наприклад GitHub).

`git init`

- Commit - збереження поточного стану файлів у репозиторії. Кожен коміт має унікальний ідентифікатор (хеш) і опис змін.

`git commit`

- Branch - незалежна лінія розробки. За замовчуванням створюється гілка master. Нові гілки дозволяють працювати над різними функціональними можливостями окремо.

`git branch <branch>`

`git checkout -b <branch>`

`git switch -c <branch>`

- Merge - об'єднання змін з однієї гілки в іншу. Якщо зміни конфліктують (змінюють один і той самий рядок у файлі), виникає конфлікт злиття, який потрібно вирішити вручну.
`git merge <branch>`
- Індекс (Staging area) - проміжна область, куди додаються файли командою `git add` перед комітом.
`git add .`
- `git log --all--graph` - перегляд історії комітів у вигляді графа.

Хід роботи

1. Створення нової папки `mkdir l`, перехід у неї `cd l` і ініціалізація git-репозиторію `git init`.
2. Зроблено перший порожній коміт із повідомленням `init` для подальшої роботи з репозиторієм. `git commit --allow-empty -m "init"`
3. Створено гілку **b1**. `git branch b1` – створює гілку але не переходить у неї. Створено гілку **b2**. `git switch -c b2` – створює гілку та переходить у неї. За допомогою команди `git branch` перевірено кількість гілок та статус.
4. Створено текстовий файл `echo "2" > f2.txt`, на гілку `b1` було додано файл `f2.txt`. За допомогою команди `git add f2.txt` файл було додано у staging area, та відповідно після цього було виконано коміт `git commit -m "sds"`
5. У гілці **b2** створено файл `f2.txt` із вмістом «3» та закомічено. В цьому випадку можна використати команду `git add .` оскільки все що було створено призначено для гілки `b2`.
`echo "3" > f2.txt`
`git add .`
`git commit -m "dds"`
Також було виведено граф для перевірки правильності
6. Виконано злиття гілки **b1** у **b2**, `git merge b1` під час якого виник конфлікт у файлі `f2.txt`.
7. Конфлікт було вирішено вручну: відкрито файл у редакторі `vim`, залишено потрібний варіант та збережено `:wq`.
8. За допомогою команди `git log --all --graph` виведено гілки та їх статус у вигляді графа.

```

PS C:\Users\vovak> mkdir 1

Directory: C:\Users\vovak


Mode                LastWriteTime         Length Name
----                -
d-----          20.09.2025     9:19             1

PS C:\Users\vovak> cd 1
PS C:\Users\vovak\1> git init
Initialized empty Git repository in C:/Users/vovak/1/.git/
PS C:\Users\vovak\1> git commit --allow-empty -m 'init'
[master (root-commit) 9234ffff] init

PS C:\Users\vovak\1> git branch b1
PS C:\Users\vovak\1> git switch -c b2
Switched to a new branch 'b2'
PS C:\Users\vovak\1> git branch
  b1
* b2
  master

PS C:\Users\vovak\1> git switch b1
A       f2.txt
Switched to branch 'b1'
PS C:\Users\vovak\1> echo '2'>f2.txt
PS C:\Users\vovak\1> git add f2.txt
PS C:\Users\vovak\1> git status
On branch b1
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   f2.txt

PS C:\Users\vovak\1> git commit -m 'sds'
[b1 46d7d2e] sds
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 f2.txt
PS C:\Users\vovak\1> git switch b2
Switched to branch 'b2'
PS C:\Users\vovak\1> echo '3'>f2.txt
PS C:\Users\vovak\1> git add .
PS C:\Users\vovak\1> git commit -m 'dds'
[b2 3cb8c8f] dds
 2 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 f1.txt
 create mode 100644 f2.txt
PS C:\Users\vovak\1> git branch
  b1
* b2
  master

PS C:\Users\vovak\1> git log --all --graph
* commit 3cb8c8f296d086dc335abe66a6198cba712ca055 (HEAD -> b2)
  Author: neevkps <145159763+neevkps@users.noreply.github.com>
  Date:   Sat Sep 20 16:01:10 2025 +0300

    dds

* commit 46d7d2eda5fe9c3b6550d7cd54b799c15996844a (b1)
  / Author: neevkps <145159763+neevkps@users.noreply.github.com>
  Date:   Sat Sep 20 16:00:22 2025 +0300

    sds

* commit 9234ffffa8dc7ce2bb3b4aa6987a570ecdb0f8875 (master)
  Author: neevkps <145159763+neevkps@users.noreply.github.com>
  Date:   Sat Sep 20 09:20:14 2025 +0300

    init
PS C:\Users\vovak\1> git merge b1
warning: Cannot merge binary files: f2.txt (HEAD vs. b1)
Auto-merging f2.txt
CONFLICT (add/add): Merge conflict in f2.txt
Automatic merge failed; fix conflicts and then commit the result.

PS C:\Users\vovak\1> git add f2.txt
PS C:\Users\vovak\1> git merge --continue
[b2 639b8e0] Merge branch 'b1' into b2
PS C:\Users\vovak\1> git log --all --graph
* commit 639b8e079660a74ac07e816fb8fcb29789019218 (HEAD -> b2)
  \ Merge: 3cb8c8f 46d7d2e
  | Author: neevkps <145159763+neevkps@users.noreply.github.com>
  | Date:   Sat Sep 20 16:02:43 2025 +0300
  |
  | Merge branch 'b1' into b2
  |
  * commit 46d7d2eda5fe9c3b6550d7cd54b799c15996844a (b1)
  | Author: neevkps <145159763+neevkps@users.noreply.github.com>
  | Date:   Sat Sep 20 16:00:22 2025 +0300
  |
  | sds
  |
  * commit 3cb8c8f296d086dc335abe66a6198cba712ca055
  / Author: neevkps <145159763+neevkps@users.noreply.github.com>
  Date:   Sat Sep 20 16:01:10 2025 +0300

    dds

* commit 9234ffffa8dc7ce2bb3b4aa6987a570ecdb0f8875 (master)
  Author: neevkps <145159763+neevkps@users.noreply.github.com>
  Date:   Sat Sep 20 09:20:14 2025 +0300

    init

```

Висновок: у ході виконання лабораторної роботи я ознайомився з основами роботи системи контролю версій Git.