

**Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»**



ЗВІТ

**про виконання практичної роботи №2
з дисципліни
«Аналіз програмного забезпечення»**

Виконав: ст. гр. 121-22-3

Коновалов Валентин Володимирович

Перевірив: доцент Мінєєв О.С.

Дніпро

2025

Контрольні питання

1. Git - це розподілена система контролю версій (VCS), що дозволяє відстежувати зміни у файлах проєкту, особливо в коді, спрощуючи командну розробку, керування версіями та повернення до попередніх станів.
2. Репозиторій - це місце де зберігається весь проєкт, де Git відстежує зміни у файлах та дозволяє керувати версіями. Це дає можливість зберігати історію змін, повертатися до попередніх версій коду і співпрацювати з іншими розробниками, синхронізуючи зміни з віддаленим репозиторієм.
3. Переваги Git - це швидка й зручна розподілена система контролю версій, яка підходить як для маленьких проєктів, так і для великих команд. Вона підтримує нелінійну розробку та працює швидше за багато інших систем, бо зберігає знімки всього репозиторію, а не списки змін у файлах. Також Git надійний, тому що у разі збою сервера код можна відновити з локальної копії, а у разі втрати локальних даних, то із віддаленого репозиторію.
4. На GIT в основному використовується мова C, а також Bourne Shell, Tcl, Perl.
5. Щоб створити репозиторій, треба у порожній або існуючій папці проєкту скористатися командою “**git init**”
6. Команда для видалення гілки: “**git branch -d <назва гілки>**”
7. Система контролю версій - це система, що записує зміни у файл або набір файлів протягом деякого часу, це дає можливість повернутися до попередніх версій.
8. Для виправлення несправного комміту можна використовувати ці команди:

“git add <файл>” - Щоб виправити помилку в повідомленні коміту або додати файл.

“git commit –amend” - Щоб змінити коментар або додати файли до існуючого коміту.

“git reset –hard HEAD^” - Щоб видалити останній коміт.

“git reset –hard HEAD~2” - Щоб видалити останні два коміти. Можна збільшити кількість, щоб видалити ще більшу кількість комітів.

9. Щоб дізнатися, чи гілку вже об'єднано в master, можна використовувати цей порядок команд:

1. **“git checkout master”**
2. **“git pull”**
3. **“git branch –merged”**