Міністерство освіти і науки України

Львівський національний університет імені Івана Франка

Факультет електроніки та комп’ютерних технологій

Звіт про виконання лабораторної роботи №5

**“** **ВЕРСІОНУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ. ЛОКАЛЬНА СИСТЕМА КОНТРОЛЮ ВЕРСІЙ”**

Виконав студент групи ФеП-41:

Радь О. Ю.

Перевірив:

доцент Франів В. А.

**Львів 2024**

**Мета роботи:** Мета лабораторної роботи полягає у вивченні концепції версіонування програмного забезпечення, розумінні важливості та переваг використання систем контролю версій, а також у освоєнні практичних навичок створення локальних репозиторіїв та ефективного користування ними базуючись на технології Git.

**Введення**

Локальна система контролю версій (Local Version Control System, LVCS) — це тип системи, яка допомагає користувачам відстежувати зміни у файлах на одному комп'ютері. Вона зберігає версії файлів у локальній файловій системі, дозволяючи користувачам легко керувати змінами, повертатися до попередніх версій та аналізувати історію змін.

**Основні характеристики локальних систем контролю версій:**

1. **Локальне зберігання**: Всі версії файлів зберігаються на локальному комп'ютері, без необхідності підключення до віддаленого сервера.
2. **Історія змін**: Користувач може бачити історію змін для кожного файлу, включаючи інформацію про те, коли і ким були внесені зміни.
3. **Відкат змін**: Можливість повернення до попередніх версій файлів у випадку помилок або необхідності аналізу.
4. **Контроль версій**: Система створює резервні копії файлів при кожній зміні, дозволяючи зберігати різні версії одного і того ж файлу.

**Переваги локальних систем контролю версій:**

* **Простота використання**: Зазвичай легкі у налаштуванні та використанні, не потребують серверної інфраструктури.
* **Швидкість**: Оскільки всі дані зберігаються локально, операції з контролем версій виконуються швидко.
* **Контроль та автономність**: Користувач має повний контроль над своїми даними і не залежить від зовнішніх сервісів.

**Недоліки локальних систем контролю версій:**

* **Обмеження у співпраці**: Складно організувати спільну роботу декількох користувачів, оскільки система працює лише на одному комп'ютері.
* **Ризик втрати даних**: Втрата або пошкодження локального комп'ютера може призвести до втрати всіх версій файлів.
* **Масштабованість**: Не підходять для великих проектів з численними учасниками, де потрібна координація між багатьма користувачами.

**Хід роботи**

1. Освоїв теоретичний матеріал.
2. Встановив систему контролю версій
3. Ініціалізував локальний git репозиторій (рис. 3.1).

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Рис. 3.1. Ініціалізація пустого репозиторію

1. Створив декілька нових файлів та добавив їх до гіт репозиторію (рис. 4.1).

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Рис. 4.1. Додавання файлів до локального репозиторію

1. Створив перший коміт для добавлених файлів з повідомленням: This is my first commit (рис. 5.1)

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Рис. 5.1. Створення коміту з повідомленням

1. Здійснив зміни у файлах та зробив другий коміт (рис. 6.1)

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Рис. 6.1. Створення другого коміту

1. Об’єднав два попередні коміти в один (рис. 7.1)

Зображення, що містить текст, знімок екрана, Шрифт, програмне забезпечення

Автоматично згенерований опис

Рис. 7.1. Об’єднання першого та другого коміту в один

**Висновок:** Git є потужним інструментом для управління версіями, що дозволяє розробникам зберігати історію змін, працювати над різними гілками проекту паралельно, а також легко інтегруватися з іншими інструментами DevOps.

Використання Git сприяє кращій організації командної роботи, дозволяючи кожному учаснику бачити внесок колег, відслідковувати зміни та ефективно вирішувати конфлікти.

Завдяки лабораторній роботі ми здобули практичні навички роботи з Git, які є необхідними для будь-якого сучасного розробника.