**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ**

**„КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ”**

**НАВЧАЛЬНО-НАУКОВИЙ КОМПЛЕКС**

**„ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ”**

Лабораторна робота №1

з курсу «Проектування інформаційних систем» на тему:

“Системи контролю версій SVN, GIT.”

Виконала:

студентка 4 курсу

групи ДА-62

Слєпцова О.О.

Київ 2019

**Мета роботи:** за допомогою системи контролю версій завантажити коди програми у репозіторій. Відтворити типовий цикл розробки програмного забезпечення з використанням системи контролю версій.

**Завдання:**

1. Обрати безкоштовну систему репозиторія для системи контролю версіями,

наприклад projectlocker, або інш.

2. Встановити клієнтське безкоштовне програмне забезпечення для роботи с системою

контролю версій (GIT, SVN clients).

3. Протягом роботи над лабораторними роботами 2-6 використовувати систему

контролю версіями.

4. Описати цикл розробки програмного забезпечення з використанням системи

контролю версій.

**Короткі теоретичні відомості:**

**Система керування версіями** (СКВ, англ. source code management, SCM) — програмний інструмент для керування версіями одиниці інформації: вихідного коду програми, скрипту, веб-сторінки, веб-сайту, 3D-моделі, текстового документу тощо.

**Система керування версіями** — інструмент, який дозволяє одночасно, не заважаючи один одному, проводити роботу над груповими проектами.

Системи керування версіями зазвичай використовуються при розробці програмного забезпечення для відстеження, документування та контролю над поступовими змінами в електронних документах: у сирцевому коді застосунків, кресленнях, електронних моделях та інших документах, над змінами яких одночасно працюють декілька людей.

Кожна версія позначається унікальною цифрою чи літерою, зміни документу занотовуються. Зазвичай також зберігаються дані про автора зробленої зміни та її час.

Інструменти для контролю версій входять до складу багатьох інтегрованих середовищ розробки.

Існують два основні типи систем керування версіями: з централізованим сховищем та розподіленим.

Система збереження історії редагувань статей, що застосовується у Вікіпедії є прикладом системи керування версіями.

**Git** — розподілена система керування версіями файлів та спільної роботи. Проект створив Лінус Торвальдс для управління розробкою ядра Linux, а сьогодні підтримується Джуніо Хамано ([англ.](https://uk.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D1%96%D0%B9%D1%81%D1%8C%D0%BA%D0%B0_%D0%BC%D0%BE%D0%B2%D0%B0) *Junio C. Hamano*). Git є однією з найефективніших, надійних і високопродуктивних систем керування версіями, що надає гнучкі засоби нелінійної розробки, що базуються на відгалуженні і злитті гілок. Для забезпечення цілісності історії та стійкості до змін заднім числом використовуються криптографічні методи, також можлива прив'язка цифрових підписів розробників до тегів і комітів.

**Subversion** — централізована система. Дані зберігаються в єдиному сховищі. При збереженні нових версій використовується дельта-компресія, тобто система знаходить відмінності нової версії від попередньої і записує тільки їх, уникаючи непотрібного дублювання даних. Сховище може розташовуватися на локальному диску або на мережевому сервері. До локального сховища клієнт Subversion звертається безпосередньо. Для доступу до віддаленого сервера може використовуватися власний мережевий протокол або стандартний протокол WebDAV, підтримуваний за допомогою спеціального модуля для веб-сервера Apache.

### **Централізовані системи контролю версій**

*Централізована система контролю версії* (клієнт-серверна) — система, дані в якій зберігаються в єдиному «серверному» сховищі. Весь обмін файлами відбувається з використанням центрального сервера. Є можливість створення та роботи з локальними репозиторіями (робочими копіями).

Переваги:

* загальна нумерація версій;
* дані знаходяться на одному сервері;
* можлива реалізація функції блокування файлів;
* можливість керування доступом до файлів;

Недоліки:

* оновлення робочої копії та збереження змін можливе лише за наявності мережевого з'єднання з основним сервером.

### **Розподілені системи контролю версії**

Розподілена система контролю версії (англ. *Distributed Version Control System*, DVCS) — система, яка використовує замість моделі клієнт-сервер, розподілену модель зберігання файлів. Така система не потребує сервера, адже всі файли знаходяться на кожному з комп'ютерів.

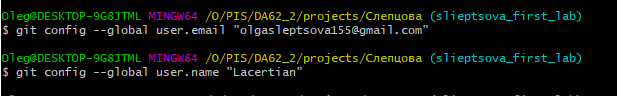
Переваги:

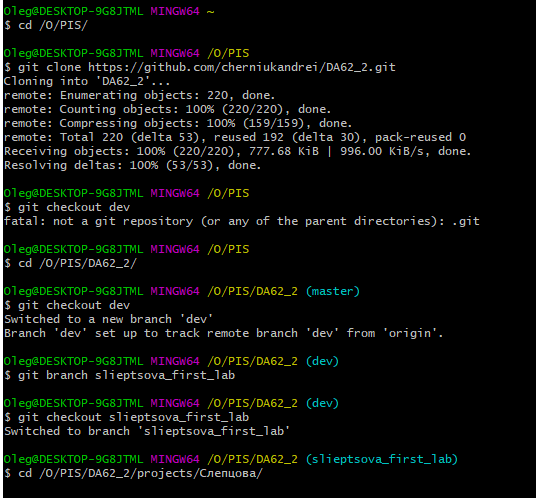
* Кожний з розробників працює зі своїм власним репозитарієм;
* Рішення щодо злиття гілок приймається керівником проекту;
* Немає потреби в мережевому з'єднанні;

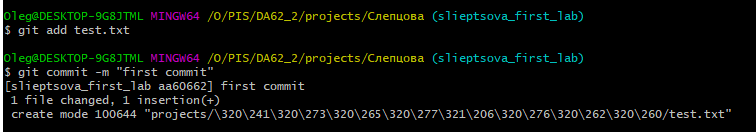
Недоліки:

* не можливо контролювати доступ до файлів;
* відсутня загальна нумерація версій файлів;
* потрібно значно більше дискового простору на сервері для зберігання всіх гілок як глобальних, так й локальних;
* відсутня можливість блокування файлів.

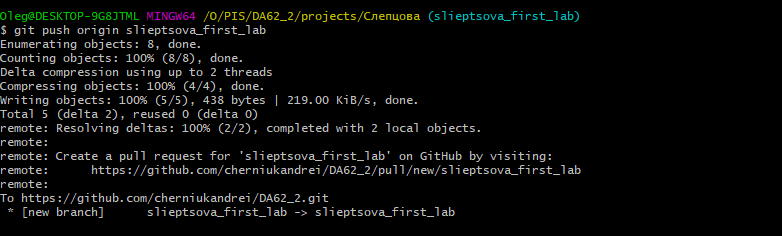
**Основні команди Git:**

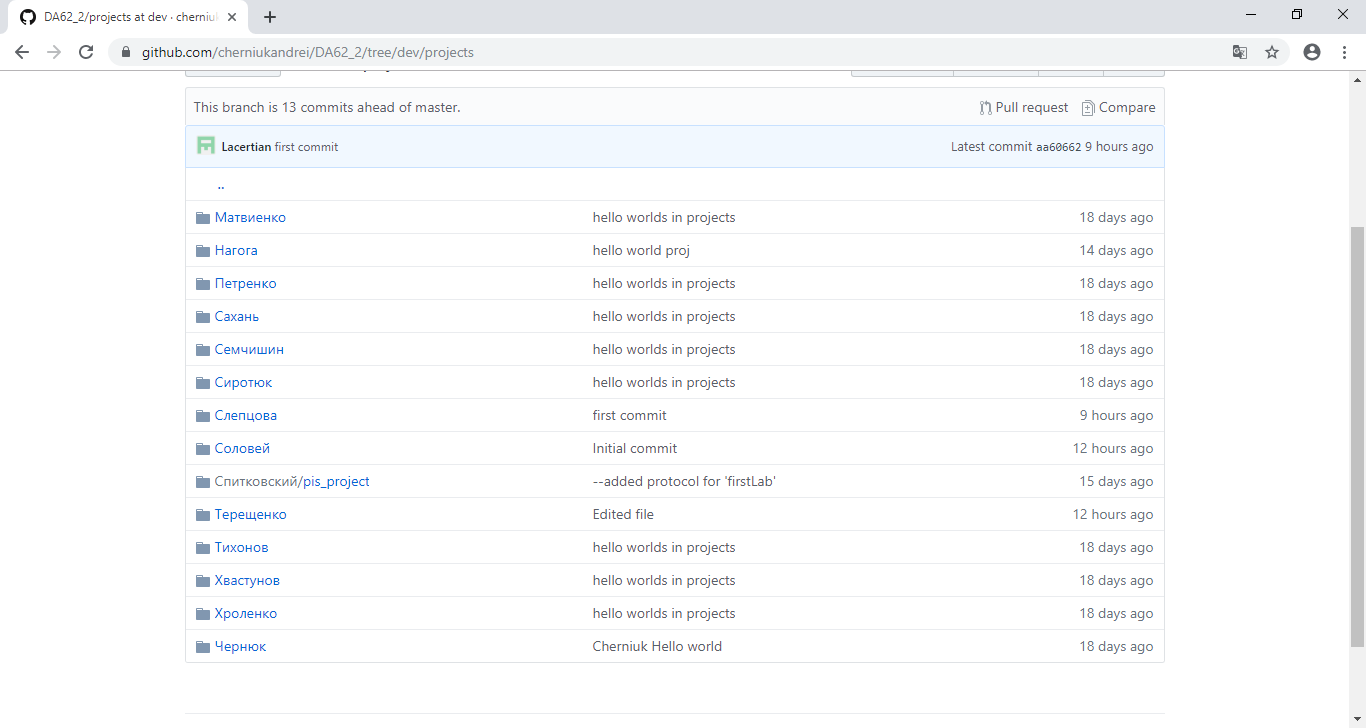
1. Створення нового репозиторія - git init
2. Перевірка статусу – git status
3. Переміщення - git add
4. Внесення коміту – git commit
5. Підключення до дистанційного репозиторія – git remote add
6. Завантаження на сервер – git push
7. Клонування репозиторія – git clone
8. Отримання змін з серверу – git pull
9. Створення нової гілки – git branch
10. Перехід між гілками – git checkout
11. З'єднання гілок – git merge





Пушим на серевер:





**Висновки:** у ході роботи було набуто навичок роботи з системою контролю версій - Git, опановано основні команди clone, branch, checkout, push, commit.