Міністерство освіти і науки України

Запорізький національний технічний університет

кафедра програмних засобів

ЗВІТ

З лабораторної роботи №1

З дисципліни «Якість програмного забезпечення» з теми:

3 Варіант

Виконав:

студент групи КНТ-415 Д.А. Рязанов

Прийняв: Т.І. Каплієнко

2018

**Виконання роботи**

**Варіант 3**

Ініціалізувати папку з майбутнім проектом в git. Створити файл .gitignore з описом ігнорованих файлів (\*. [oa], \* ~). Користувач вводить свої дані, створює папку проекту, створює в ній файл із завданнями. Виконує комміт для фіксації внесених змін. Створює нову гілку, відкриває файл і виконує завдання, виконує комміт. Повертається в гілку master. Переглядає історію коммітів використовуючи одностроковий формат. Вносить зміни у файл, виконує комміт. Переглядає історію, використовуючи різні формати контролю відображення записів.

1. Створюємо репозиторій для проекта

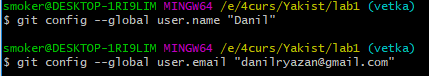


2. Створюємо файл .gitignore з описом ігнорованих файлів





3. Користувач вводить свої дані (ім’я и email)



4 Створює в ній файл із завданнями



5. Додає усі файли до репозиторію

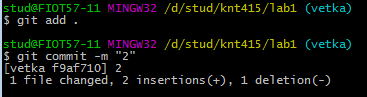


6. Виконує коміт



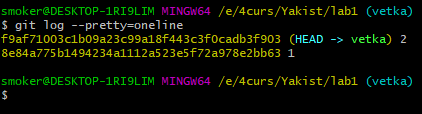
7. Створює нову гілку

8. Виконує своє завдання, яке прописано керівником команди

9. Додає усі файли до репозиторію і робить коміт

10. Повертається в гілку master

11. Переглядає історію коммітів використовуючи одностроковий формат



**Теорія по Git**

**ОСНОВНІ ХАРАКТЕРИСТИКИ GIT**

Переваги:

1. Надійна система порівняння ревізій та перевірки коректності даних, заснована на алгоритмі хеширувания SHA1 (Secure Hash Algorithm 1).

2. Гнучка система розгалуження проектів і злиття гілок між собою.

3. Наявність локального репозиторію,

4. Висока продуктивність і швидкість роботи.

5. Зручний і інтуїтивно зрозумілий набір команд.

6. Безліч графічних оболонок, що дозволяють швидко і якісно вести роботи з Git'ом.

7. Можливість робити контрольні точки.

8. Широка розповсюдженність, легка доступність і якісна документація.

9. Гнучкість системи.

10. Універсальний мережевий доступ з використанням протоколів http, ftp, rsync, ssh та ін.

Недоліки

1. Unix - орієнтованість.

2. Можливі (але надзвичайно низькі) збіги хеш - коду відмінних за змістом ревізій.

3. Не відстежуються зміни окремих файлів

4. При початковому (першому) створенні сховища та синхронізації його з іншими розробниками, буде потрібно досить тривалий час для скачування даних, особливо, якщо проект великий, так як потрібно скопіювати на локальний комп'ютер весь репозиторій.

**АРХІТЕКТУРА GIT**

• SHA1 (Secure Hash Algorithm 1)

• Об’єкти Git

• Blob (Binary Large Object)

• Дерево (Tree)

• Reference

• Гілка (Head, Branch)

• Тег (tag) /мітка

• light tag

• annotated tag