Workcase 1 ЗВІТ

1. 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.

Команда git add додає вміст робочого каталогу індекс (staging area) для наступного комміта. За замовчанням git commit використовує лише цей індекс, так що ви можете використовувати git add для складання зліпка наступного комміту.

Команда git status показує стан файлів у робочому каталозі та індексі: які файли змінені, але не додані в індекс; які очікують на комміт в індексі. На додаток до цього виводяться підказки про те, як змінити стан файлів.

Команда git difftool просто запускає зовнішню утиліту порівняння для показу відмінностей у двох деревах, на випадок, якщо ви хочете використовувати щось відмінне від вбудованого переглядача git diff.

Команда git commit бере всі дані, додані до індексу за допомогою git add, і зберігає їх зліпок у внутрішній базі даних, а потім зрушує покажчик поточної гілки на цей зліпок.

Команда git rm використовується в Git для видалення файлів з індексу та робочої копії. Вона схожа на git add з тим винятком, що вона видаляє, а не додає файли для наступного комміту.

Команда git mv — це всього лише зручний спосіб перемістити файл, а потім виконати git add для нового файлу і git rm для старого.

2. Коміт у [git](https://uk.wikipedia.org/wiki/Git" \o "Git) — це об'єкт, що містить посилання на:

-знімок стану що записаний раніше в індекс

-метадані автора

-коментарі

-нуль чи більш вказівників на інші коміти,що є прямими батьками даного коміту.

Залежно від ситуації, у коміта може бути така кількість батьків:

* 0 для першого коміта
* 1 для звичайного коміта
* декілька для коміта, що отримується в результаті злиття двох чи більше гілок.

Виконуючи команду git commit, Git створюєте коміт. При цьому Git вираховує контрольну суму для кожного підкаталогу і зберігає ці об'єкти-дерева в Git-репозиторій. Після цього Git створює об'єкт для коміту, в якому є метадані і вказівник на об'єкт-дерево. Таким чином, Git зможе відтворити поточний стан, коли в цьому буде необхідність.

Коміт дозволяє відслідковувати зміни таким чином що, ми наприклад бачимо коментарій від розробника в команді і можемо відслідковувати зміни наприклад в коді.

Висновки:

Ми визначили, що сайт GitHub дуже ефективний для робіт на віддаленій відстані. Він надійний, ефективний, використовує криптографічні методи, а також стійкий до змін.

Роботу виконували: Яременко Олексій, Руда Вікторія, Чуркін Дмитро.