**Системы контроля версий. Основы работы в Git**

**Цель работы:** ознакомление с основными возможностями системы контроля

версий Git.Задание

3. Проверьте установленную версию и зафиксируйте ее в отчете:

4. Получите справку по Git. Ознакомьтесь со списком командам. Какой командой для получения справки вы воспользовались? Описание поместите в отчет.

5. Создайте локальный репозиторий. Опишите последовательность действий в отчете.

6. Выполните конфигурацию гит. Для этого настройте конфигурационный файл, используя: $ git config --global user.name "XXX" $ git config --global user.email [XXX@XXX.com](mailto:XXX@XXX.com)

7. Проверить настройки конфигурационного файла командой: git config –list

8. Инициализируйте локальный репозиторий в рабочей папке (см. лекцию), где будет он будет размещен. Исследуйте изменения, которые произошли в папке. Зафиксируйте их в отчете.

9. С помощью блокнота создайте в папке текстовый файл Hello.txt.

10.Научитесь получать информацию о статусе (status) репозитория. Какая информация отображается в терминальном окне GitBash? Прокомментируйте вывод команды в отчете. 11.Проиндексируйте файл Hello.txt Какие изменения произошли в репозитории?

12.Зафиксируйте изменения индексированных файлов в репозитории. Какие изменения произошли в репозитории? Как Git информирует об успешном создании нового коммита? В какую ветку добавлен файл?

13.Просмотрите журнал с помощью команды git log

14.Научитесь получать информацию о проиндексированных и неиндексированных файлах. Для этого измените существующий файл Hello.txt, добавьте в репозиторий новый файл и исследуйте состояние репозитория. Зафиксируйте изменения в репозитории

. 15.Научитесь пользоваться командами управления и фиксации изменений (add, commit, status, log).

16.Ознакомиться с шаблонами .gitignore и настройте этот файл для вашего репозитория.

Работа с ветками в локальном репозитории

1. Создайте новую ветку (branch), например, test и переключитесь (checkout) в нее.

2. Куда указывает HEAD?

3. Просмотрите содержимое всех файлов в папке проекта, с которыми вы работали.

4. Создайте новый текстовый файл test.html и зафиксируйте его (commit).

5. Просмотрите список всех веток и определите текущую ветку.

6. Выполните слияние веток (merge).

7. По очереди отредактируйте файл test.html в ветках master и test. Выполните commit. Выполните слияние в одной и второй ветках. Разрешите ситуацию конфликта слияний.

8. Научитесь пользоваться командами Git для работы с ветками.

Работа с удаленным репозиторием

1. Создайте аккаунт на <https://github.com>

2. Изучите настройки профиля (по адресу <https://github.com/settings/>)

3. Создайте свой первый репозиторий с именем «OPI\_lab»

4. Добавьте в репозиторий Readme-файл, используя web-интерфейс.

5. Внесите в него информацию о себе по шаблону из лекции. Отформатируйте текст.

6. Сохраните файл и зафиксируйте изменения в репозитории.

7. Изучите обновленный репозиторий: просмотрите историю коммитов, изменений файлов и т.п. 8. Выполните клонирование удаленного репозитория в Git.

9. Просмотрите конфигурационный файл локального репозитория.

10.Внесите изменения в файл README.md и зафиксируйте изменения в Git.

11.Просмотрете в Git список настроенных удаленных репозиториев.

12.Отправьте изменения в удаленный репозиторий (команда Push).

13.Исследуйте, как изменился удаленный репозиторий.

14.Добавьте в удаленный репозиторий ваш отчет по лабораторной работе через web-интерфейс. 15.Зафиксируйте изменения.

16. Получите изменения из удаленного репозитория в локальный (команда fetch).

17.Научитесь работать с удаленным и локальным репозиториями, используя сценарий и команды, приведенные в лекции.

Совместная работа с удаленным репозиторием (выполнить задание можно, создав свой второй аккаунт в GitHub).

1. Разбейтесь по парам и создайте форк репозитория напарника.

2. Внесите изменения в своей ветке форка (например, добавив какой-нибудь файл).

3. Создайте запрос (pull-реквест) в репозиторий напарника, предложив свои изменения.

4. После принятия ваших изменений и их слияния в исходный репозиторий его владельцем, заберите в свой форк последние изменения.

5. Объясните последовательность действий и произошедшие изменения.