РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2

* дисциплина: Архитектура компьютера*

Студент: Замбалова Дина Владимировна

Группа: НПИбд-01-22

МОСКВА

2023 г.

Цель работы
- Изучить идеологию и применение средств контроля версий.
- Освоить умения по работе с git.

Теоретические сведения
Системы контроля версий (Version Control System, VCS) применяются при работе нескольких человек над одним проектом. Обычно основное дерево проекта хранится в локальном или удалённом репозитории, к которому настроен доступ для участников проекта. При внесении изменений в содержание проекта система контроля версий позволяет их фиксировать, совмещать изменения, произведённые разными участниками проекта, производить откат к любой более ранней версии проекта, если это требуется.

В классических системах контроля версий используется централизованная модель, предполагающая наличие единого репозитория для хранения файлов. Выполнение большинства функций по управлению версиями осуществляется специальным сервером. Участник проекта (пользователь) перед началом работы посредством определённых команд получает нужную ему версию файлов. После внесения изменений, пользователь размещает новую версию в хранилище. При этом предыдущие версии не удаляются из центрального хранилища и к ним можно вернуться в любой момент. Сервер может сохранять не полную версию изменённых файлов, а производить так называемую дельта-компрессию — сохранять только изменения между последовательными версиями, что позволяет уменьшить объём хранимых данных.

Системы контроля версий поддерживают возможность отслеживания и разрешения конфликтов, которые могут возникнуть при работе нескольких человек над одним файлом. Можно объединить (слить) изменения, сделанные разными участниками (автоматически или вручную), вручную выбрать нужную версию, отменить изменения вовсе или заблокировать файлы для изменения. В зависимости от настроек блокировка не позволяет другим пользователям

получить рабочую копию или препятствует изменению рабочей копии файла средствами файловой системы ОС, обеспечивая таким образом, привилегированный доступ только одному пользователю, работающему с файлом.

Системы контроля версий также могут обеспечивать дополнительные, более гибкие функциональные возможности. Например, они могут поддерживать работу с несколькими версиями одного файла, сохраняя общую историю изменений до точки ветвления версий и собственные истории изменений каждой ветви. Кроме того, обычно доступна информация о том, кто из участников, когда и какие изменения вносил. Обычно такого рода информация хранится в журнале изменений, доступ к которому можно ограничить.

В отличие от классических, в распределённых системах контроля версий центральный репозиторий не является обязательным.

Среди классических VCS наиболее известны CVS, Subversion, а среди распределённых — Git, Bazaar, Mercurial. Принципы их работы схожи, отличаются они в основном синтаксисом используемых в работе команд.

Задание
- Создать базовую конфигурацию для работы с git.
- Создать ключ SSH.
- Создать ключ PGP.
- Настроить подписи git.
- Зарегистрироваться на Github.
- Создать локальный каталог для выполнения заданий по предмету.

Выполнение лабораторной работы
Установка программного обеспечения. Устанавливаю git: dnf install git и gh (рис. [-@fig:fig1])
![Установка git, gh](https://i.ibb.co/rwwZJdX/42.png){ #fig:fig1 width=70% }

Базовая настройка git. Зададим имя и email владельца репозитория. Настроим utf-8 в выводе сообщений git. Зададим имя начальной ветки (будем называть её master). Параметр autocrlf. Параметр safecrlf.(рис. [-@fig:fig2]).
![Базовая настройка git](https://i.ibb.co/RhZ9sFB/43.png){ #fig:fig2 width=70% }

Создаю ключи ssh по алгоритму rsa с ключём размером 4096 бит (рис. [-@fig:fig3]) и по алгоритму ed25519 (рис. [-@fig:fig4])
![Создние ключа ssh](https://i.ibb.co/GpSbnqb/44.png){ #fig:fig43 width=70% }
![Создние ключа ssh](https://i.ibb.co/2KyCxGY/27.png){ #fig:fig4 width=70% }

Создаю ключи pgp. Генерируем ключ. Из предложенных опций выбираем:
тип RSA and RSA;
размер 4096;
выберите срок действия; значение по умолчанию — 0 (срок действия не истекает никогда).
GPG запросит личную информацию, которая сохранится в ключе:
Имя (не менее 5 символов).
Адрес электронной почты.

При вводе email убеждаюсь, что он соответствует адресу, используемому на GitHub.
Комментарий. Можно ввести что угодно или нажать клавишу ввода, чтобы оставить это поле пустым. (рис. [-@fig:fig5]
![Создание ключа pgp](https://i.ibb.co/J52xDhC/29.png){ #fig:fig5 width=70% }

Учетгая запись github была создана в прошлом семестре.

Добавление PGP ключа в GitHub. Выводим список ключей и копируем отпечаток приватного ключа. Отпечаток ключа — это последовательность байтов, используемая для идентификации более длинного, по сравнению с самим отпечатком ключа. Копирую мой сгенерированный PGP ключ в буфер обмена (рис. [-@fig:fig6])
![Добавление PGP ключа в GitHub](https://i.ibb.co/kSYYKK9/38.png){ #fig:fig6 width=70% }

Перехожу в настройки GitHub (https://github.com/settings/keys), нажмаю на кнопку New GPG key и вставляю полученный ключ в поле ввода (рис. [-@fig:fig7])
![Добавление PGP ключа в GitHub](https://i.ibb.co/64600jZ/41.png){ #fig:fig7 width=70% }

Настройка автоматических подписей коммитов git. Используя введёный email, указываю Git применять его при подписи коммитов (рис. [-@fig:fig8])
![Настройка автоматических подписей коммитов git](https://i.ibb.co/k6zVFBW/45.png){ #fig:fig8 width=70% }

Настройка gh. Для начала необходимо авторизоваться. Утилита задаст несколько наводящих вопросов. Авторизоваться можно через броузер (рис. [-@fig:fig9])
![Настройка gh](https://i.ibb.co/9bZKhRq/46.png){ #fig:fig9 width=70% }

Создание репозитория курса на основе шаблона (рис. [-@fig:fig10])
![Создание репозитория курса на основе шаблона](https://i.ibb.co/QDrV2t9/47.png){ #fig:fig10 width=70% }

Настройка каталога курса. Перехожу в каталог курса. Удаляю лишние файлы. Создаю необходимые каталоги. Отправляю файлы на сервер (рис. [-@fig:fig11]), (рис. [-@fig:fig12])
![Настройка каталога курса](https://i.ibb.co/mR8Dm8G/35.png){ #fig:fig11 width=70% }
![Настройка каталога курса](https://i.ibb.co/6NDmvPQ/36.png){ #fig:fig12 width=70% }

Выводы
Я изучила идеологию и применение средств контроля версий. Освоила умения по работе с git.