

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: Лопатченко Полина Андреевна

Студ. билет :1032253529

Группа: НКАбд-04-25

МОСКВА

2025 г.

Содержание

1. Цель работы.....	4
2. Выполнение лабораторной работы.....	5-14
2.1 Настройка github.....	5
2.2 Базовая настройка git	6
2.3 Создание SSH-ключа	7
2.4 Создание рабочего пространства.....	8
2.5 Создание репозитория курса на основе шаблона	9
2.6 Настройка каталога курса	10-11
2.7. Задание для самостоятельной работы	12
2.8. Контрольные вопросы для самопроверки.....	13-14
3. Вывод.....	15

Список иллюстраций

1.1 Создание учетной записи на сайте github.....	5
2.1 Ввод всех команд в терминал.....	6
3.1 Генерация ключей в терминале	7
3.2 Сохранение ключа в github	7
4.1 Создание каталога.....	8
5.1 Создание репозитория.....	9
5.2 Переход в каталог.....	9
5.3 Клонирование репозитория.....	9
6.1 Переход в каталог курса.....	10
6.2 Создание необходимых каталогов	10
6.3 Отправление файлов	10
6.4 Проверка переноса изменений в локальный репозиторий	11
7.1 Подгрузила лабораторные на github	12

1. Цель работы.

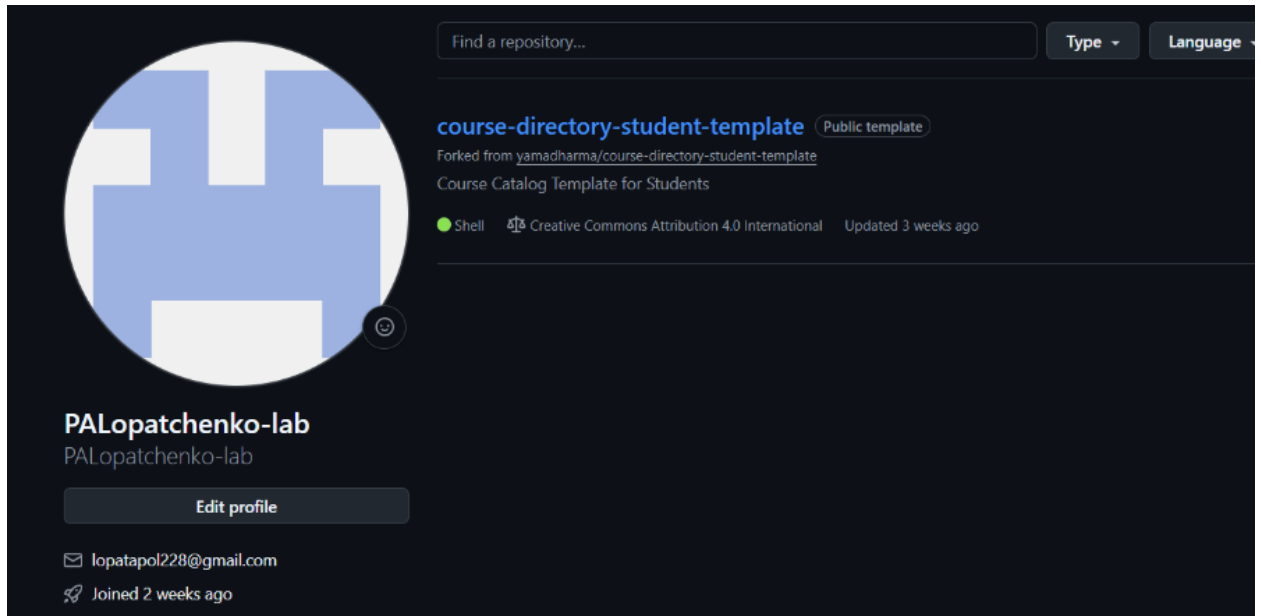
Целью работы является изучение идеологии и применения средств контроля версий, приобретение практических навыков по работе с системой контроля версий git.

2.1 Выполнение лабораторной работы.

2.1 Настройка github.

- Создайте учётную запись на сайте <https://github.com/> и заполните основные данные.

Создала учетную запись на сайте github.



1.1 Создание учетной записи на сайте github

2.2 Базовая настройка git.

- Сначала сделаем предварительную конфигурацию git. Откройте терминал и введите следующие команды, указав имя и e-mail владельца репозитория:
`git config --global user.name "<Name Surname>"`
`git config --global user.email "<work@mail>"`
- Настроим utf-8 в выводе сообщений git:
`git config --global core.quotepath false`
- Зададим имя начальной ветки (будем называть её master):
`git config --global init.defaultBranch master`
- Параметр autocrlf:
`git config --global core.autocrlf input`
- Параметр safecrlf:
`git config --global core.safecrlf warn`

Ввела в терминал команды, указав свои данные репозитория.

```
P.A.Lopatchenko@polina-VirtualBox:~$ git config --global user.name "<PALopatchenko-lab>"
P.A.Lopatchenko@polina-VirtualBox:~$ git config --global user.email "<lopatapol228@gmail.com>"
P.A.Lopatchenko@polina-VirtualBox:~$ git config --global core.quotepath false
P.A.Lopatchenko@polina-VirtualBox:~$ git config --global init.defaultBranch master
P.A.Lopatchenko@polina-VirtualBox:~$ git config --global core.autocrlf input
P.A.Lopatchenko@polina-VirtualBox:~$ git config --global core.safecrlf warn
P.A.Lopatchenko@polina-VirtualBox:~$
```

2.1 Ввод всех команд в терминал

2.3 Создание SSH-ключа

- Для последующей идентификации пользователя на сервере репозитория необходимо сгенерировать пару ключей (приватный и открытый):
`ssh-keygen -C "Имя Фамилия <work@mail>"`

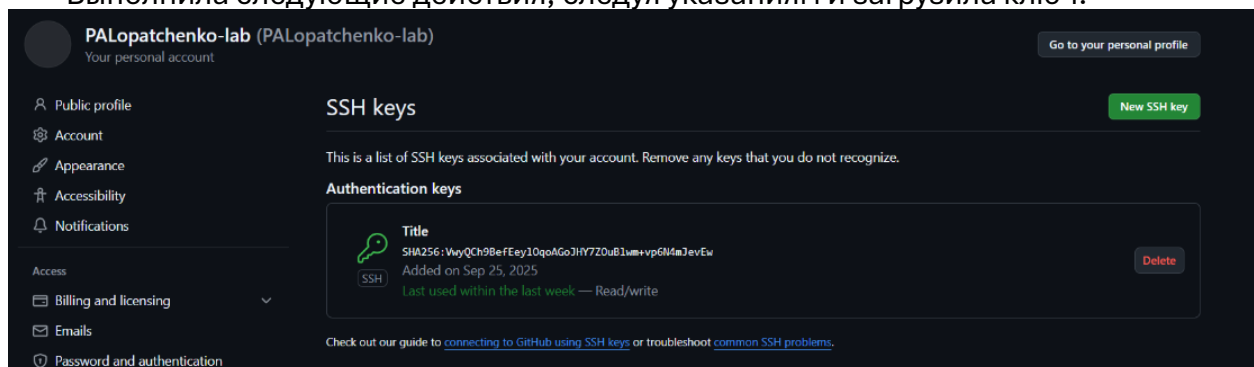
Сгенерировала необходимую пару ключей.

```
P.A.Lopatchenko@polina-VirtualBox:~$ ssh-keygen -C "Polina Lopatchenko <lopatapol228@gmail.com>"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/P.A.Lopatchenko/.ssh/id_ed25519):
/home/P.A.Lopatchenko/.ssh/id_ed25519 already exists.
Overwrite (y/n)? y
Enter passphrase for "/home/P.A.Lopatchenko/.ssh/id_ed25519" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/P.A.Lopatchenko/.ssh/id_ed25519
Your public key has been saved in /home/P.A.Lopatchenko/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:VwyQCh9BefEeyl0qoAGoJHY7Z0uBlwm+vp6N4mJevEw Polina Lopatchenko <lopatapol228@gmail.com>
The key's randomart image is:
+--[ED25519 256]--+
|      .000+.      |
| .    . o o . o   |
|++ + o + + o     |
|*.* = o. = o     |
|.o.O. S o        |
| =oo. . o        |
| ..E .           |
|+.B .            |
|BO.+             |
+-----[SHA256]-----+
```

3.1 Генерация ключей в терминале

- Далее необходимо загрузить сгенерированный открытый ключ. Для этого следует зайти на сайт <http://github.org/> под своей учётной записью и перейти в меню Setting . После этого выбрать в боковом меню SSH and GPG keys и нажать кнопку New SSH key . Копируем из локальной консоли ключ в буфер обмена
`cat ~/.ssh/id_rsa.pub | xclip -sel clip`. Вставляем ключ в появившееся на сайте поле и указываем для ключа имя (Title).

Выполнила следующие действия, следуя указаниям и загрузила ключ.



3.2 Сохранение ключа в github

2.4 Создание рабочего пространства и репозитория курса на основе шаблона.

- При выполнении лабораторных работ следует придерживаться структуры рабочего пространства. Рабочее пространство по предмету располагается в следующей иерархии:

~/work/study/

 <учебный год>/

 <название предмета>/

 <код предмета>/

- Каталог для лабораторных работ имеет вид labs.
Каталоги для лабораторных работ имеют вид lab<номер>, например: lab01, lab02 и т.д. Название проекта на хостинге git имеет вид: study_<учебный год>_<код предмета>. Например, для 2025–2026 учебного года и предмета «Архитектура компьютера» (код предмета arch-pc) название проекта примет следующий вид: study_2025–2026_arch-pc. Откройте терминал и создайте каталог для предмета «Архитектура компьютера»:
mkdir -p ~/work/study/2023-2024/"Архитектура компьютера"

Создала каталог для предмета «Архитектура компьютеров»

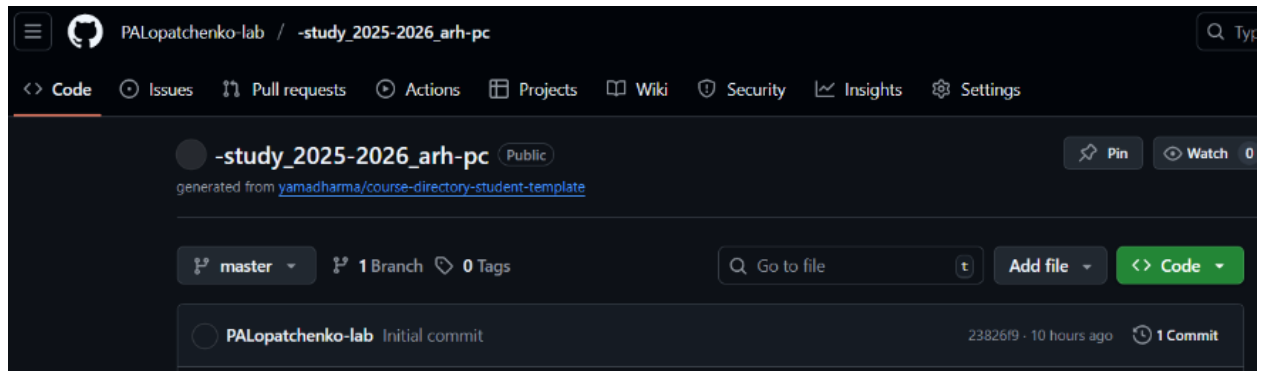
```
P.A.Lopatchenko@polina-VirtualBox:~$ mkdir -p ~/work/study/2025-2026/"Архитектура компьютера"  
P.A.Lopatchenko@polina-VirtualBox:~$
```

4.1 Создание каталога

2.5 Создание репозитория курса на основе шаблона

- Репозиторий на основе шаблона можно создать через web-интерфейс github. Перейдите на страницу репозитория с шаблоном курса <https://github.com/yamadharma/course-directory-student-template>. Далее выберите Use this template . В открывшемся окне задайте имя репозитория (Repository name) study_2025–2026_arh-pc и создайте репозиторий (кнопка Create repository from template).

Задала имя репозитория и создала репозиторий



5.1 Создание репозитория

- Откройте терминал и перейдите в каталог курса:
`cd ~/work/study/2025–2026/"Архитектура компьютера"`

Открыла терминал и перешла в каталог курса.

```
P.A.Lopatchenko@polina-VirtualBox: ~/work/study/2025-2026/Архитектура компьютера
P.A.Lopatchenko@polina-VirtualBox: ~/work/study/2025-2026/Архитектура компьютера$ git clone --recursive git@github.com:PALopatchenko-lab/-study_2025-2026_arh-pc.git
```

5.2 Переход в каталог

- Клонируйте созданный репозиторий:
`git clone --recursive git@github.com:<user_name>/study_2025–2026_arh-pc.git`
arch-pc

Клониrowала созданный репозиторий

```
P.A.Lopatchenko@polina-VirtualBox: ~/work/study/2025-2026/Архитектура компьютера$ git clone --recursive git@github.com:PALopatchenko-lab/-study_2025-2026_arh-pc.git
Клонирование в «-study_2025-2026_arh-pc»...
Enter passphrase for key '/home/P.A.Lopatchenko/.ssh/id_ed25519':
remote: Enumerating objects: 38, done.
remote: Counting objects: 100% (38/38), done.
remote: Compressing objects: 100% (36/36), done.
remote: Total 38 (delta 1), reused 27 (delta 1), pack-reused 0 (from 0)
Получение объектов: 100% (38/38), 23.45 КиБ | 228.00 КиБ/с, готово.
Определение изменений: 100% (1/1), готово.
Подмодуль «template/presentation» (https://github.com/yamadharma/academic-presentation-markdown-template.git) зарегистрирован по пути «template/presentation»
Подмодуль «template/report» (https://github.com/yamadharma/academic-laboratory-report-template.git) зарегистрирован по пути «template/report»
Клонирование в «/home/P.A.Lopatchenko/work/study/2025-2026/Архитектура компьютера/-study_2025-2026_arh-pc/template/presentation»...
remote: Enumerating objects: 161, done.
remote: Counting objects: 100% (161/161), done.
remote: Compressing objects: 100% (111/111), done.
remote: Total 161 (delta 60), reused 142 (delta 41), pack-reused 0 (from 0)
Получение объектов: 100% (161/161), 2.65 МБ | 227.00 КиБ/с, готово.
Определение изменений: 100% (60/60), готово.
Клонирование в «/home/P.A.Lopatchenko/work/study/2025-2026/Архитектура компьютера/-study_2025-2026_arh-pc/template/report»...
remote: Enumerating objects: 221, done.
remote: Counting objects: 100% (221/221), done.
remote: Compressing objects: 100% (152/152), done.
remote: Total 221 (delta 98), reused 180 (delta 57), pack-reused 0 (from 0)
Получение объектов: 100% (221/221), 765.46 КиБ | 177.00 КиБ/с, готово.
Определение изменений: 100% (98/98), готово.
Submodule path 'template/presentation': checked out '6efd5c4ee78e4456caff3dc7062cfcad26058ca6'
Submodule path 'template/report': checked out '89a9622199b4df88227b9b3fa3d4714c85f68dd2'
P.A.Lopatchenko@polina-VirtualBox: ~/work/study/2025-2026/Архитектура компьютера$
```

5.3 Клонирование репозитория

2.6 Настройка каталога курса

- Перейдите в каталог курса:
`cd ~/work/study/2025-2026/"Архитектура компьютера"/arch-pc`

Перешла в каталог курса.

```
P.A.Lopatchenko@polina-VirtualBox:~/work/study/2025-2026/Архитектура компьютера$ cd ~/work/study/2025-2026/"Архитектура компьютера"/arch-pc
P.A.Lopatchenko@polina-VirtualBox:~/work/study/2025-2026/Архитектура компьютера/arch-pc$
```

6.1 Переход в каталог курса

- Создайте необходимые каталоги:
`echo arch-pc > COURSE`
`make prepare`

Создала необходимые каталоги

```
P.A.Lopatchenko@polina-VirtualBox:~/work/study/2025-2026/Архитектура компьютера/arch-pc$ echo arch-pc > COURSE
P.A.Lopatchenko@polina-VirtualBox:~/work/study/2025-2026/Архитектура компьютера/arch-pc$ make prepare
```

6.2 Создание необходимых каталогов

- Отправьте файлы на сервер:
`git add .`
`git commit -am 'feat(main): make course structure'`
`git push`

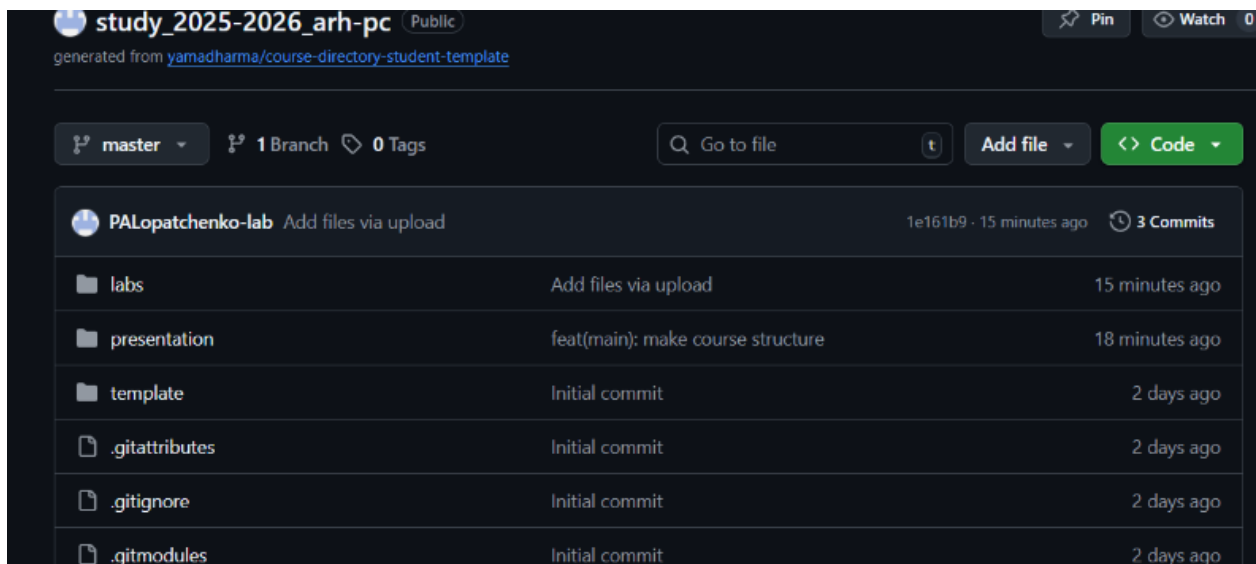
Отправила файлы на сервер.

```
P.A.Lopatchenko@polina-VirtualBox:~/work/study/2025-2026/Архитектура компьютера/arch-pc$ git push
Enter passphrase for key '/home/P.A.Lopatchenko/.ssh/id_ed25519':
Перечисление объектов: 67, готово.
Подсчет объектов: 100% (67/67), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (52/52), готово.
Запись объектов: 100% (64/64), 700.31 КиБ | 4.46 МиБ/с, готово.
Total 64 (delta 22), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (22/22), completed with 1 local object.
To github.com:PALopatchenko-lab/study_2025-2026_arh-pc.git
 23826f9..bbf8cdb master -> master
P.A.Lopatchenko@polina-VirtualBox:~/work/study/2025-2026/Архитектура компьютера/arch-pc$
```

6.3 Отправление файлов

- Проверьте правильность создания иерархии рабочего пространства в локальном репозитории и на странице github.

Проверила правильность создания иерархии рабочего пространства в локальном репозитории и на странице github.



6.4 Проверка переноса изменений в локальный репозиторий

2.7 Задания для самостоятельной работы.

- Создайте отчет по выполнению лабораторной работы в соответствующем каталоге рабочего пространства (labs/lab02/report).

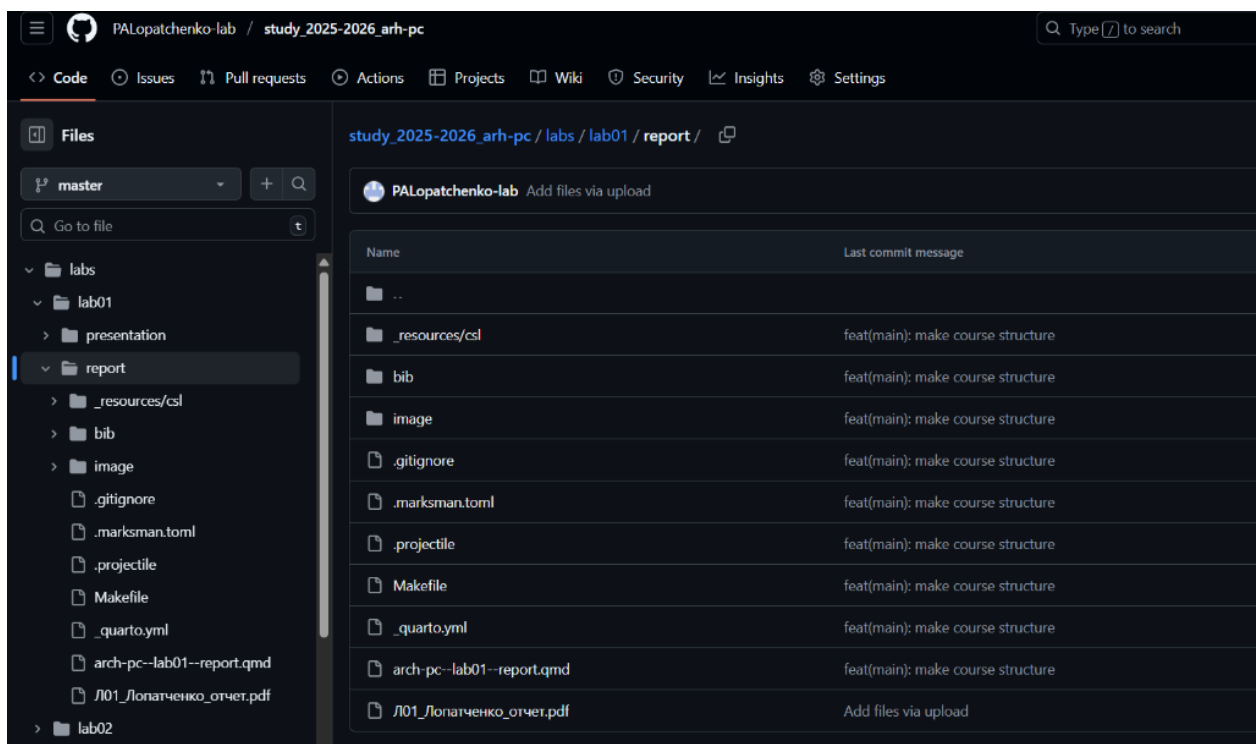
Создала отчет по выполнению лабораторной работы.

- Скопируйте отчеты по выполнению предыдущих лабораторных работ в соответствующие каталоги созданного рабочего пространства.

Скопировала отчеты по выполнению предыдущих лабораторных работ соответствующие каталоги созданного рабочего пространства.

- Загрузите файлы на github.

Загрузила файлы на github.



7.1 Подгрузила лабораторные на github

2.8. Контрольные вопросы для самопроверки.

1. Что такое системы контроля версий (VCS) и для решения каких задач они предназначаются?

Системы контроля версий (VCS) — это программы, которые помогают отслеживать и управлять изменениями в файлах (чаще всего в коде). Они позволяют сохранять историю изменений, возвращаться к предыдущим версиям, работать с разными вариантами разработки и удобно сотрудничать в команде.

2. Объясните следующие понятия VCS и их отношения: хранилище, commit, история, рабочая копия.

- Хранилище (репозиторий) — место, где сохраняются все версии файлов и история изменений.

- Commit (коммит) — снимок состояния файлов на определённый момент, фиксирующий изменения и добавляющий их в историю.

- История — последовательность всех коммитов, показывающая всю эволюцию проекта.

- Рабочая копия — текущие файлы на компьютере, с которыми работаете.

3. Что представляют собой и чем отличаются централизованные и децентрализованные VCS? Приведите примеры VCS каждого вида.

- Централизованные VCS имеют один центральный сервер с хранилищем, к которому подключаются пользователи для загрузки и выгрузки изменений. Пример: Subversion (SVN), CVS

- Децентрализованные VCS каждый пользователь имеет полное локальное копирование репозитория со всей историей и может работать автономно. Пример: Git, Mercurial

4. Опишите действия с VCS при единоличной работе с хранилищем.

- Создать репозиторий.
- Работать с файлами в рабочей копии.
- Отслеживать изменения и делать коммиты, сохраняя версии.
- При необходимости откатываться к предыдущим коммитам.

5. Опишите порядок работы с общим хранилищем VCS.

- Клонировать удалённый репозиторий (получить копию).
- Делать изменения в рабочей копии.
- Делать коммиты локально.
- Отправлять изменения в удалённый репозиторий (push).
- Получать изменения других участников (pull).

6. Каковы основные задачи, решаемые инструментальным средством git?

- Отслеживание изменений в проекте.
- Работа с несколькими версиями (ветками).
- Возможность работать офлайн.
- Обеспечение совместной работы через удалённые репозитории.
- Быстрое переключение между состояниями проекта.

7. Назовите и дайте краткую характеристику командам git.

- git init — создать новый репозиторий.
- git clone — скопировать удалённый репозиторий.
- git add — добавить файлы в индекс для следующего коммита.
- git commit — сохранить изменения локально
- git status — показать текущее состояние рабочих файлов.
- git push — отправить изменения в удалённый репозиторий.
- git pull — получить и объединить изменения из удалённого репозитория.
- git checkout — переключаться между ветками или коммитами.

8. Приведите примеры использования при работе с локальным и удалённым репозиториями.

Локальный репозиторий

- git init (создать)
- git add файл (добавить файл)
- git commit -m "Комментарий" (сохранить)

Удалённый репозиторий (для работы с уже готовыми файлами)

- git clone url_репозитория (клон)
- git pull (обновление)
- git push (отправить на изменение)

3 Вывод

В ходе проведенной лабораторной работы мы изучили идеологию и применение средств контроля версий, приобрели практические навыки по работе с системой контроля версий git.