

# **Лабораторная работа №2**

**Система контроля версий Git**

Борисенкова София Павловна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
<b>3</b>	<b>Задание для самостоятельной работы</b>	<b>11</b>
<b>4</b>	<b>Выводы</b>	<b>12</b>

## Список иллюстраций

2.1	Рисунок 1. Настройка параметров git . . . . .	6
2.2	Рисунок 2. Генерация ssh ключа . . . . .	7
2.3	Рисунок 3. Вкладка настроек GitHub . . . . .	7
2.4	Рисунок 4. создание каталога «Архитектура компьютера» . . . . .	8
2.5	Рисунок 5. Клонирование репозитория . . . . .	9
2.6	Рисунок 6. Отправка репозитория на GitHub . . . . .	10

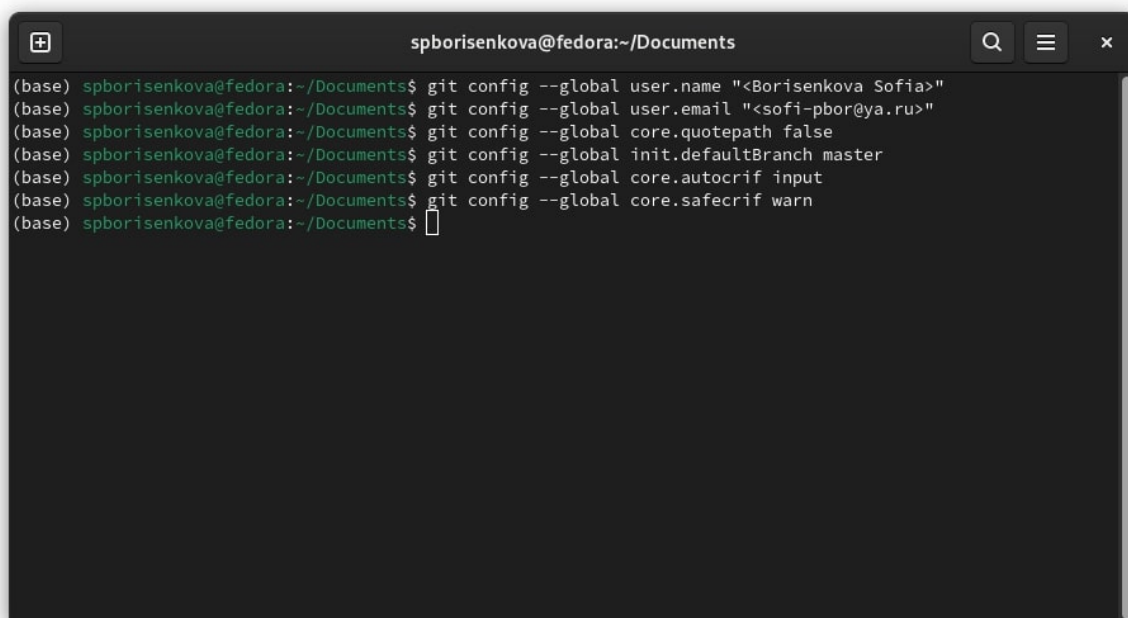
## **Список таблиц**

# 1 Цель работы

Целью работы является изучить идеологию и применение средств контроля версий. Приобрести практические навыки по работе с системой git.

## 2 Выполнение лабораторной работы

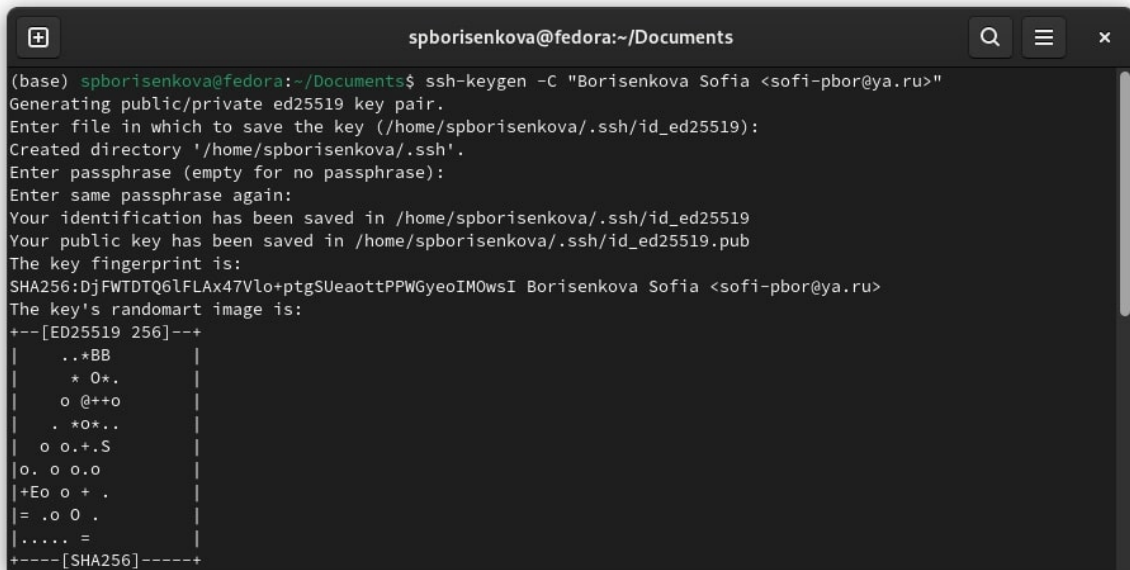
Сначала сделаем предварительную конфигурацию git. Откроем терминал и введём следующие команды, указав имя и email владельца репозитория и другие параметры. (рис. 1)

A screenshot of a terminal window titled 'spborisenkova@fedora:~/Documents'. The window contains a series of git configuration commands entered in a shell. The commands are: 'git config --global user.name "<Borisenkova Sofia>"', 'git config --global user.email "<sofi-pbor@ya.ru>"', 'git config --global core.quotepath false', 'git config --global init.defaultBranch master', 'git config --global core.autocrlf input', and 'git config --global core.safecrlf warn'. The prompt '(base) spborisenkova@fedora:~/Documents\$' is visible at the start of each line, and a cursor is at the end of the last line.

```
(base) spborisenkova@fedora:~/Documents$ git config --global user.name "<Borisenkova Sofia>"
(base) spborisenkova@fedora:~/Documents$ git config --global user.email "<sofi-pbor@ya.ru>"
(base) spborisenkova@fedora:~/Documents$ git config --global core.quotepath false
(base) spborisenkova@fedora:~/Documents$ git config --global init.defaultBranch master
(base) spborisenkova@fedora:~/Documents$ git config --global core.autocrlf input
(base) spborisenkova@fedora:~/Documents$ git config --global core.safecrlf warn
(base) spborisenkova@fedora:~/Documents$
```

Рис. 2.1: Рисунок 1. Настройка параметров git

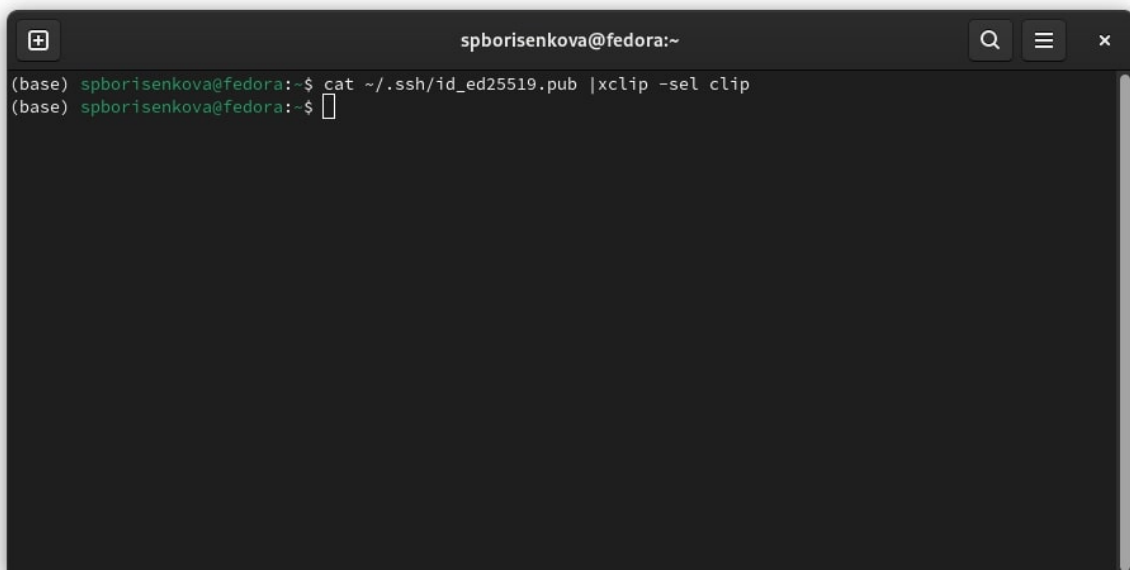
Для того, чтобы сервер мог идентифицировать пользователя, необходимо сгенерировать ssh ключи. Для этого мы введем следующую команду. (рис. 2)

A terminal window titled 'spborisenkova@fedora:~/Documents'. The user runs the command 'ssh-keygen -C "Borisenkova Sofia <sofi-pbor@ya.ru>"'. The terminal shows the process of generating an ed25519 key pair, saving it to '/home/spborisenkova/.ssh/id\_ed25519', and displaying the key's fingerprint and a randomart image.

```
(base) spborisenkova@fedora:~/Documents$ ssh-keygen -C "Borisenkova Sofia <sofi-pbor@ya.ru>"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/spborisenkova/.ssh/id_ed25519):
Created directory '/home/spborisenkova/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/spborisenkova/.ssh/id_ed25519
Your public key has been saved in /home/spborisenkova/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:DjFWTDtQ6lFLAx47Vlo+ptgSUEaottPPWgyeoIM0wsI Borisenkova Sofia <sofi-pbor@ya.ru>
The key's randomart image is:
+--[ED25519 256]--+
|  ..*BB          |
|   * O*         |
|  o @++o        |
|   *O*..        |
| o o.+S         |
|o. o o.o        |
|+Eo o + .       |
|= .o O .        |
|.....=         |
+----[SHA256]-----+
```

Рис. 2.2: Рисунок 2. Генерация ssh ключа

Затем скопируем следующей командой ключ в буфер обмена и вставляем его в соответствующее поле на github. (рис. 3)

A terminal window titled 'spborisenkova@fedora:~'. The user runs the command 'cat ~/.ssh/id\_ed25519.pub |xclip -sel clip'. The terminal shows the command being executed and a cursor on the next line.

```
(base) spborisenkova@fedora:~$ cat ~/.ssh/id_ed25519.pub |xclip -sel clip
(base) spborisenkova@fedora:~$
```

Рис. 2.3: Рисунок 3. Вкладка настроек GitHub

Откроем терминал и создадим каталог для предмета «Архитектура компьюте-

ра». (рис. 4)

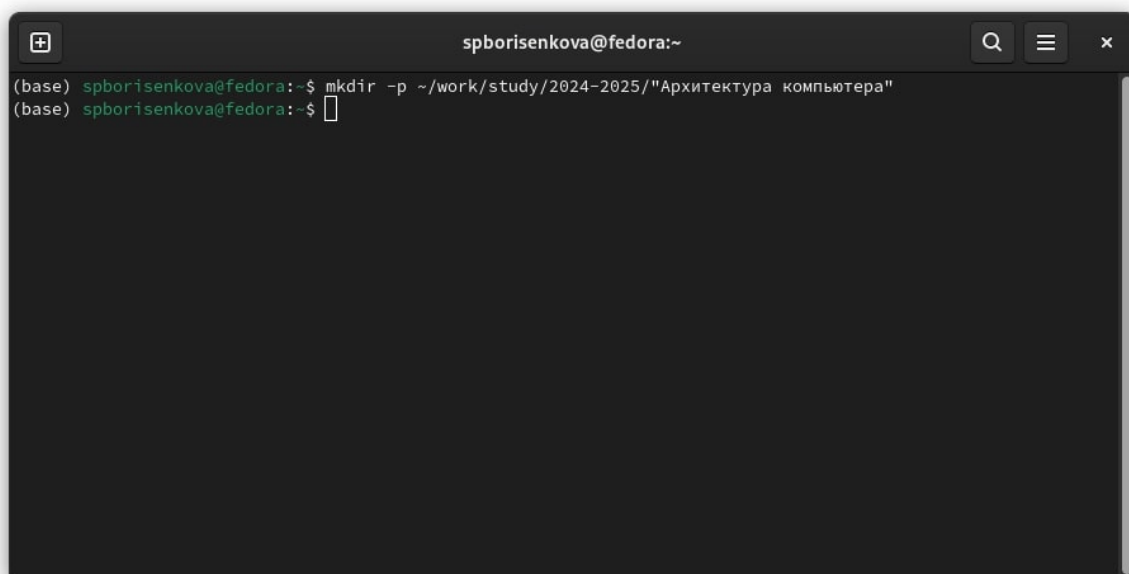
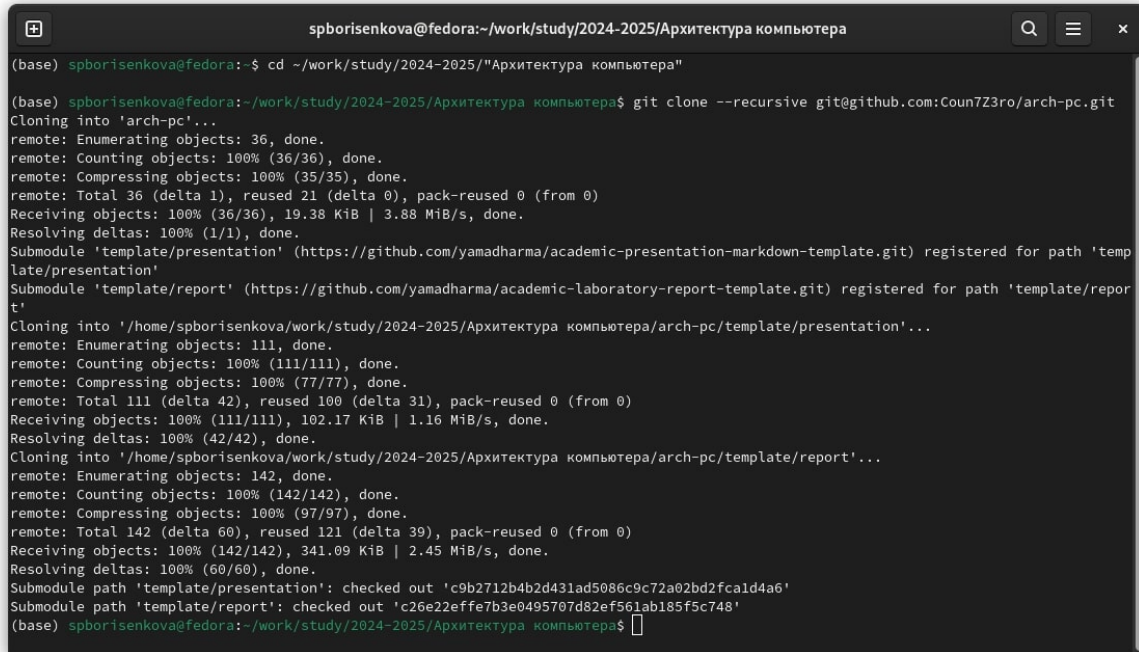


Рис. 2.4: Рисунок 4. создание каталога «Архитектура компьютера»

Создадим репозиторий. Будем создавать его на основе шаблона, который находится по следующему адресу: <https://github.com/yamadharm/course-directory-student-template> Нажимаем на кнопку “Use this template”, и в предложенных опциях выбираем “Create new repository”. Задаём имя репозитория. Он будет называться arch-рс. Далее нам нужно клонировать репозиторий на наш компьютер. Для этого перейдем в папку “Архитектура компьютера” и клонируем репозиторий. (рис. 5)





```
(base) spborisenkova@fedora:~/work/study/2024-2025/Архитектура компьютера$ cd ~/work/study/2024-2025/Архитектура компьютера
(base) spborisenkova@fedora:~/work/study/2024-2025/Архитектура компьютера$ git clone --recursive git@github.com:Coun7Z3ro/arch-pc.git
Cloning into 'arch-pc'...
remote: Enumerating objects: 36, done.
remote: Counting objects: 100% (36/36), done.
remote: Compressing objects: 100% (35/35), done.
remote: Total 36 (delta 1), reused 21 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (36/36), 19.38 KiB | 3.88 MiB/s, done.
Resolving deltas: 100% (1/1), done.
Submodule 'template/presentation' (https://github.com/yamadharma/academic-presentation-markdown-template.git) registered for path 'template/presentation'
Submodule 'template/report' (https://github.com/yamadharma/academic-laboratory-report-template.git) registered for path 'template/report'
Cloning into '/home/spborisenkova/work/study/2024-2025/Архитектура компьютера/arch-pc/template/presentation'...
remote: Enumerating objects: 111, done.
remote: Counting objects: 100% (111/111), done.
remote: Compressing objects: 100% (77/77), done.
remote: Total 111 (delta 42), reused 100 (delta 31), pack-reused 0 (from 0)
Receiving objects: 100% (111/111), 102.17 KiB | 1.16 MiB/s, done.
Resolving deltas: 100% (42/42), done.
Cloning into '/home/spborisenkova/work/study/2024-2025/Архитектура компьютера/arch-pc/template/report'...
remote: Enumerating objects: 142, done.
remote: Counting objects: 100% (142/142), done.
remote: Compressing objects: 100% (97/97), done.
remote: Total 142 (delta 60), reused 121 (delta 39), pack-reused 0 (from 0)
Receiving objects: 100% (142/142), 341.09 KiB | 2.45 MiB/s, done.
Resolving deltas: 100% (60/60), done.
Submodule path 'template/presentation': checked out 'c9b2712b4b2d431ad5086c9c72a02bd2fca1d4a6'
Submodule path 'template/report': checked out 'c26e22effe7b3e0495707d82ef561ab185f5c748'
(base) spborisenkova@fedora:~/work/study/2024-2025/Архитектура компьютера$
```

Рис. 2.5: Рисунок 5. Клонирование репозитория

Теперь перейдем к настройке клонированного каталога. Для начала перейдем в него с помощью команды “cd”, удалим лишний файл с помощью команды “rm” и создадим необходимые каталоги. Теперь нам остается отправить файлы на сервер. Для этого с помощью команды git add мы добавим каталоги, которые должны отправляться на сервер. Теперь с помощью команды git commit мы сохраним изменения и укажем комментарий, в котором будет поясняться, какие изменения мы сделали. В данном случае в комментарии мы напишем, что создали структуру курса. Теперь нам осталось окончательно загрузить изменения на сервер. Для этого мы воспользуемся командой git push. (рис. 6)

```
spborisenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc
(base) spborisenkova@fedora:~/work/study/2024-2025/Архитектура компьютера$ cd arch-pc
(base) spborisenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ rm package.json
(base) spborisenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ echo arch-pc > COURSE
(base) spborisenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git add .
(base) spborisenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git commit -am 'feat(main): make course structure'
[master 4e38e16] feat(main): make course structure
2 files changed, 1 insertion(+), 14 deletions(-)
delete mode 100644 package.json
(base) spborisenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 289 bytes | 289.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:Coun7Z3ro/arch-pc.git
5cb659d..4e38e16 master -> master
(base) spborisenkova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$
```

Рис. 2.6: Рисунок 6. Отправка репозитория на GitHub

### 3 Задание для самостоятельной работы

Теперь приступим к выполнению самостоятельной работы. Для начала мы создадим файл отчета для нашей лабораторной работы в папке `labs/lab02/report`. После этого скопируем отчет по нашей предыдущей лабораторной работе в соответствующую папку созданного нами рабочего пространства, то есть в папку `labs/lab01/report`. Теперь нам осталось лишь загрузить изменения на GitHub. Воспользуемся командой `“git add .”` для того, чтобы указать, что мы хотим сохранить изменения во всех файлах, находящихся в нашем каталоге. После этого с помощью команды `“git commit”` мы укажем комментарий и сохраним изменения. (рис. 31) После этого введём команду `“git push”` для того, чтобы загрузить файлы на GitHub

## 4 Выводы

В результате выполнения лабораторной работы появились практические навыки работы с системой контроля версий Git, была произведена её первоначальная настройка в linux. Было изучено, как создавать репозитории, сохранять изменения и добавлять к ним комментарии, а также как выгружать файлы на сервер. Были приобретены навыки работы с платформой GitHub.