

Отчёт по лабораторной работе 2

Система контроля версий Git

Ермолаев Денис Николаевич НБИбд-01-24

Содержание

1	Цель работы	5
2	Порядок выполнения лабораторной работы	6
2.1	Настройка Github	6
2.2	Базовая настройка git	6
2.3	Создание SSH ключа	7
2.4	Создание рабочего пространства и репозитория курса на основе шаблона	8
2.5	Создание репозитория на основе шаблона	9
2.6	Настройка каталога курса	9
2.7	Задание для самостоятельной работы	10
3	Выводы	11

Список иллюстраций

2.1	Базовая настройка git	7
2.2	Создание SSH ключа	8
2.3	Создание рабочего пространства	9
2.4	Клонирование репозитория	9
2.5	Настройка каталога курса	10
2.6	Настройка каталога курса	10
2.7	Настройка каталога курса	10
2.8	Настройка каталога курса	10

Список таблиц

1 Цель работы

Изучить идеологию и применение систем контроля версий, приобрести практические навыки по работе с системой Git.

2 Порядок выполнения лабораторной работы

2.1 Настройка Github

Для начала я создал учётную запись на сайте GitHub по адресу <https://github.com/>. После создания аккаунта я заполнил свои основные данные, такие как имя пользователя и email. Это нужно для того, чтобы Git мог идентифицировать мои действия при работе с репозиторием. Далее я настроил SSH ключ для безопасного доступа к репозиториям. Для этого я сгенерировал пару SSH ключей с помощью команды:

```
ssh-keygen -C "Мое Имя мой_email@mail.com".
```

Полученный публичный ключ был добавлен в мой аккаунт GitHub для последующей аутентификации.

2.2 Базовая настройка git

После настройки учётной записи на GitHub, я выполнил предварительную конфигурацию git. В терминале я ввёл следующие команды для настройки имени пользователя и электронной почты:

```
git config --global user.name "Мое Имя"
```

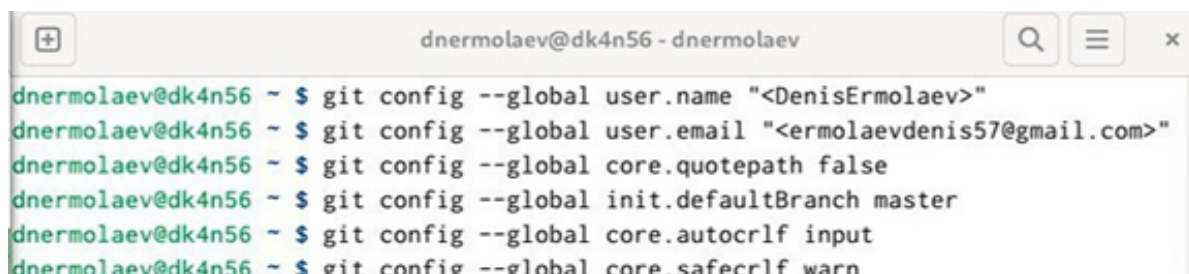
```
git config --global user.email "мой_email@mail.com"
```

Это нужно для того, чтобы каждый мой commit был подписан моими данными.

Для корректного отображения сообщений в git я также настроил параметр utf-8, введя команду:

```
git config --global core.quotepath false.
```

(рис. 2.1)

A screenshot of a terminal window with a title bar that reads "dnermolaev@dk4n56 - dnermolaev". The terminal shows a series of six git configuration commands being executed. The prompt "dnermolaev@dk4n56 ~ \$" is repeated for each command. The commands are: "git config --global user.name "<DenisErmolaev>"", "git config --global user.email "<ermolaevdenis57@gmail.com>"", "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".

```
dnermolaev@dk4n56 ~ $ git config --global user.name "<DenisErmolaev>"
dnermolaev@dk4n56 ~ $ git config --global user.email "<ermolaevdenis57@gmail.com>"
dnermolaev@dk4n56 ~ $ git config --global core.quotepath false
dnermolaev@dk4n56 ~ $ git config --global init.defaultBranch master
dnermolaev@dk4n56 ~ $ git config --global core.autocrlf input
dnermolaev@dk4n56 ~ $ git config --global core.safecrlf warn
```

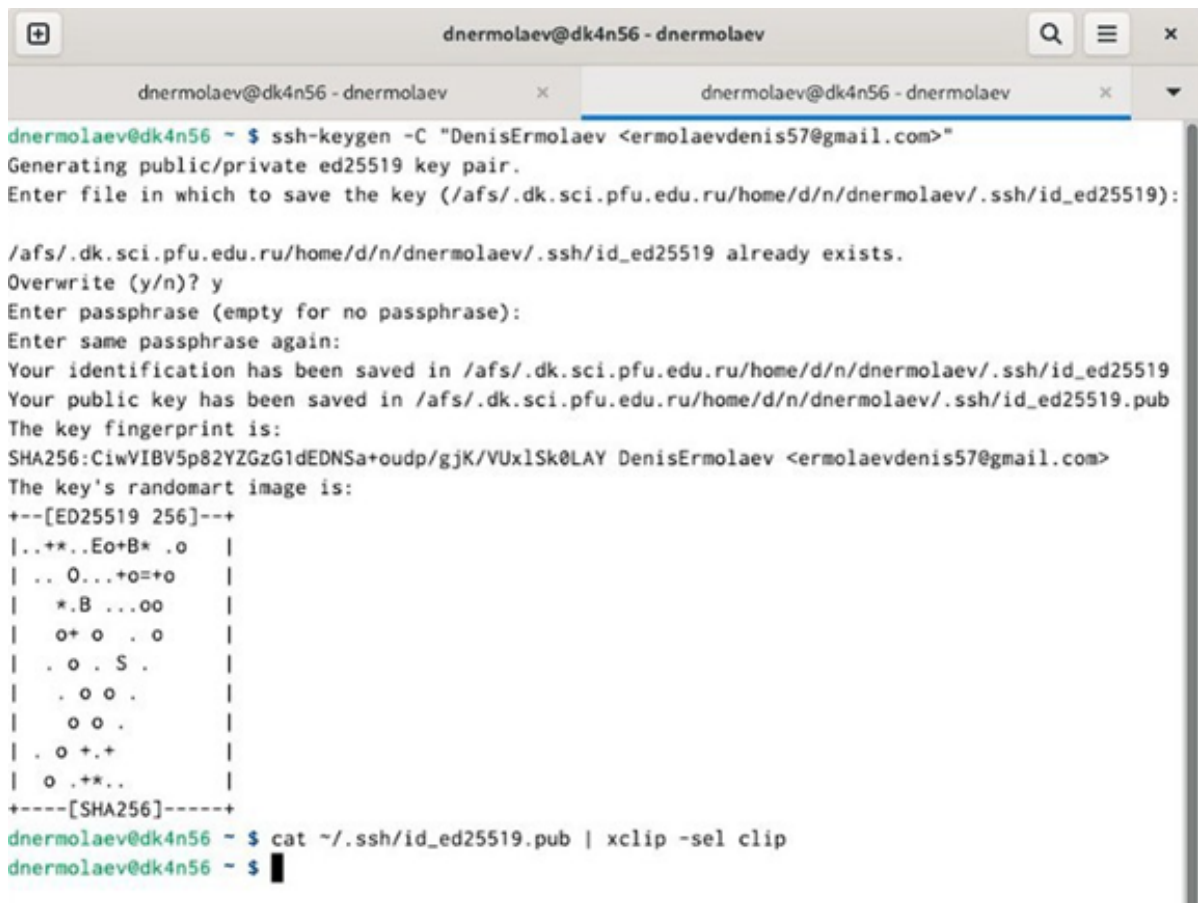
Рис. 2.1: Базовая настройка git

2.3 Создание SSH ключа

Для того чтобы безопасно подключаться к репозиторию, я сгенерировал SSH ключ с помощью команды:

```
ssh-keygen -C "Мое Имя мой_email@mail.com".
```

SSH ключ позволяет мне осуществлять безопасную аутентификацию на сервере без необходимости каждый раз вводить пароль. Полученный публичный ключ был загружен на GitHub в разделе "SSH and GPG keys". (рис. 2.2)

A screenshot of a terminal window titled 'dnermolaev@dk4n56 - dnermolaev'. The terminal shows the execution of the 'ssh-keygen' command to create an ED25519 key pair for the user 'DenisErmolaev'. The process includes prompts for a file name, passphrase, and confirmation to overwrite an existing key. It also displays the key's fingerprint and a randomart image. Finally, the public key is cat'd and copied to the clipboard using 'xclip -sel clip'.

```
dnermolaev@dk4n56 ~ $ ssh-keygen -C "DenisErmolaev <ermolaevdenis57@gmail.com>"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/afs/.dk.sci.pfu.edu.ru/home/d/n/dnermolaev/.ssh/id_ed25519):

/afs/.dk.sci.pfu.edu.ru/home/d/n/dnermolaev/.ssh/id_ed25519 already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /afs/.dk.sci.pfu.edu.ru/home/d/n/dnermolaev/.ssh/id_ed25519
Your public key has been saved in /afs/.dk.sci.pfu.edu.ru/home/d/n/dnermolaev/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:CiwVIBV5p82YZGzG1dEDNSa+oudp/gjK/VUxlSk0LAY DenisErmolaev <ermolaevdenis57@gmail.com>
The key's randomart image is:
+--[ED25519 256]--+
|..+*...Eo+B* .o |
| .. 0...+o=+o |
| *.B ...oo |
| o+ o . o |
| . o . S . |
| . o o . |
| o o . |
| . o +. + |
| o .+*.. |
+-----[SHA256]-----+
dnermolaev@dk4n56 ~ $ cat ~/.ssh/id_ed25519.pub | xclip -sel clip
dnermolaev@dk4n56 ~ $
```

Рис. 2.2: Создание SSH ключа

2.4 Создание рабочего пространства и репозитория курса на основе шаблона

Я создал рабочее пространство для лабораторной работы, следуя предложенной структуре. Каталоги были созданы с помощью команды:

```
mkdir -p ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab02/.
```

Эта структура позволяет легко управлять файлами лабораторных работ.(рис. 2.3)



Рис. 2.3: Создание рабочего пространства

2.5 Создание репозитория на основе шаблона

Для упрощения работы, я создал репозиторий на основе шаблона курса через веб-интерфейс GitHub. Я использовал готовый шаблон, который предоставил преподаватель, выбрав опцию “Use this template”. Это помогло мне быстро построить все необходимые файлы и каталоги. (рис. 2.4)

```
denisfmlaev@denisfmlaev:~/work/study/2023-2024/Архитектура компьютера $ git clone --recursive git@github.com:denisfmlaev/study_2024-2025_archpc.git archpc
Клонирование в archpc...
The authenticity of host 'github.com [148.82.121.43]' can't be established.
ED25519 key fingerprint is SHA256:4D193vuvV8u3Jh8p2icf/z1DAkUP8vYnduR40vC0qg.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
remote: Enumerating objects: 33, done.
remote: Counting objects: 100% (33/33), done.
remote: Compressing objects: 100% (33/33), done.
remote: Total 33 (delta 13), reused 18 (delta 8), pack-reused 8 (from 0)
Receiving objects: 100% (33/33), 18.81 KiB | 493.00 KiB/s, 0.00s.
Receiving submodules: 100% (1/1), done.
Receiving submodules: https://github.com/yamadharma/academic-presentation-markdown-template.git: superseeded no type <template/presentation>
Receiving submodules: https://github.com/yamadharma/academic-laboratory-report-template.git: superseeded no type <template/report>
Клонирование в /afs/.../work/study/2023-2024/Архитектура компьютера/archpc/template/presentation...
remote: Enumerating objects: 131, done.
remote: Counting objects: 100% (131/131), done.
remote: Compressing objects: 100% (131/131), done.
remote: Total 131 (delta 42), reused 88 (delta 31), pack-reused 8 (from 0)
Receiving objects: 100% (131/131), 182.17 KiB | 1.58 MiB/s, 0.00s.
Receiving submodules: 100% (42/42), done.
Клонирование в /afs/.../work/study/2023-2024/Архитектура компьютера/archpc/template/report...
remote: Enumerating objects: 142, done.
remote: Counting objects: 100% (142/142), done.
remote: Compressing objects: 100% (142/142), done.
remote: Total 142 (delta 48), reused 121 (delta 39), pack-reused 8 (from 0)
Receiving objects: 100% (142/142), 341.89 KiB | 1.92 MiB/s, 0.00s.
Receiving submodules: 100% (48/48), done.
Submodule path 'template/presentation': checked out 'c162712b4b2d431ad1886c3c72a82bd7ca1d4a6'
Submodule path 'template/report': checked out 'c26e22effe7b3e94957d7d82ef581ab185f5c748'
```

Рис. 2.4: Клонирование репозитория

2.6 Настройка каталога курса

Перейдя в каталог курса, я удалил ненужные файлы, такие как package.json, и создал необходимые каталоги для курса. Затем я выполнил первичную настройку структуры репозитория с помощью команд:

```
git add .
git commit -am "feat(main): make course structure"
git push origin master
```

Эти действия загрузили структуру курса в центральный репозиторий на GitHub.
(рис. 2.5, 2.6, 2.7, 2.8,)

```
denislaev@denis: ~/work/study/2023-2024/Архитектура компьютера $ cd arch-pc/
denislaev@denis: ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ rm package.json
denislaev@denis: ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ ls
CHANGELOG.md  config  COURSE  LICENSE  Makefile  README.en.md  README.git-f20w.md  README.md  template
```

Рис. 2.5: Настройка каталога курса

```
denislaev@denis: ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ echo arch-pc > COURSE
denislaev@denis: ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ make
Usage:
  make <target>
```

Рис. 2.6: Настройка каталога курса

```
Targets:
  list          list of courses
  prepare       Generate directories structure
  submodule     Update submodules

denislaev@denis: ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ make prepare
denislaev@denis: ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ git add .
denislaev@denis: ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ git commit -am "feat(main): make course structure"
[master 976c54c] feat(main): make course structure
 223 files changed, 50881 insertions(+), 16 deletions(-)
create mode 100644 labs/README.md
create mode 100644 labs/README.ru.md
```

Рис. 2.7: Настройка каталога курса

```
denislaev@denis: ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ git push
Печисление объектов: 37, готово.
Пакет объектов: 100% (37/37), готово.
При сканировании используется до 6 потоков
Сканирование объектов: 100% (29/29), готово.
Запись объектов: 100% (15/15), 349.28 KiB | 15.51 MiB/s, готово.
Total 35 (delta 4), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (4/4), completed with 1 local object.
To github.com:denislaev/study_2024-2025_arch-pc.git
 519be02..976c54c master -> master
denislaev@denis: ~/work/study/2023-2024/Архитектура компьютера/arch-pc $
```

Рис. 2.8: Настройка каталога курса

2.7 Задание для самостоятельной работы

Для задания по самостоятельной работе я создал отчёт по выполнению лабораторной работы и разместил его в каталоге lab02. Затем я загрузил все файлы на GitHub с помощью команды:

git push.

3 Выводы

В ходе выполнения лабораторной работы я получил практические навыки работы с системой контроля версий Git. Я освоил основные команды, настроил рабочее пространство и репозиторий, а также успешно загрузил результаты на GitHub.