

Отчет по выполнению лабораторной работы 2

**Дисциплина: Архитектура компьютеров и операционные
системы:Операционные системы**

Ефремова Полина Александровна

Содержание

1 Цель работы	5
2 Задание	6
3 Теоретическое введение	7
4 Выполнение лабораторной работы	9
4.1 Установка программного обеспечения	9
4.2 Базовая настройка Git	9
4.3 Создание ключа SSH	10
4.4 Создание gpg ключа	12
4.5 Создание репозитория курса на основе шаблона	15
5 Выводы	19
6 Список литературы	20

Список иллюстраций

4.1 Установка ПО	9
4.2 Базовая настройка Git	10
4.3 Алгоритм 4096	10
4.4 Алгоритм ed25519	11
4.5 Загрузка ключа на Github	11
4.6 GPG ключ	12
4.7 GPG ключ - настройка	13
4.8 Список ключей	13
4.9 GPG ключ - копирование	14
4.10 GPG ключ - GitHub	14
4.11 Авторизация	15
4.12 Код	15
4.13 Репозиторий	16
4.14 Добавление шаблона	16
4.15 Каталог курса	17
4.16 Необходимые каталоги	17
4.17 Сервер - загрузка	18

Список таблиц

1 Цель работы

Научиться работать с языком разметки Markdown посредством выполнения отчета лабораторной работы №2.

2 Задание

1. Настройка GitHub.
2. Базовая настройка Git.
3. Создание SSH-ключа.
4. Создание рабочего пространства и репозитория курса на основе шаблона.
5. Верификация коммитов с помощью PGP.
6. Настройка каталога курса.

3 Теоретическое введение

Система контроля версий (Version Control System, VCS) – это инструмент, используемый разработчиками программного обеспечения для управления изменениями в исходном коде и других файловых ресурсах. Системы контроля версий разработаны специально для того, чтобы максимально упростить и упорядочить работу над проектом (вне зависимости от того, сколько человек в этом участвуют). СКВ дает возможность видеть, кто, когда и какие изменения вносили; позволяет формировать новые ветви проекта, объединять уже имеющиеся; настраивать контроль доступа к проекту ; осуществлять откат до предыдущих версий. 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 (слиять) 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. Демидова А. В. 14 Архитектура ЭВМ 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. Система контроля версий Git представляет собой набор программ командной строки. Доступ к ним можно получить из терминала посредством ввода команды `git` с различными опциями. Благодаря тому, что Git является распределённой системой контроля версий, резервную копию локального хранилища можно сделать простым копированием или архивацией

4 Выполнение лабораторной работы

4.1 Установка программного обеспечения

1. С помощью команд `dnf install gh` и `dnf install git-all` устанавливаю необходимое обеспечение (рис. 4.1).

Рис. 4.1: Установка ПО

4.2 Базовая настройка Git

1. Задаю имя и email владельца репозитория, настраиваю utf-8 в выводе сообщения git, задаю имя начальной ветки а также задаю параметры autocrlf и

safecrlf (рис. 4.2).

```
Завершено!
[root@vbox ~]# git config --global user.name "Polina Efremova"
[root@vbox ~]# git config --global user.email "1132246726@pfur.ru"
[root@vbox ~]# git config --global core.quotepath flase
[root@vbox ~]# git config --global init.defaultBranch master
[root@vbox ~]# git config --global core.autocrlf true
[root@vbox ~]# git config --global core.safecrlf warn
[root@vbox ~]#
```

Рис. 4.2: Базовая настройка Git

4.3 Создание ключа SSH

- Создаю ключ по алгоритму 4096 (рис. 4.3).

```
[paefremova@vbox ~]$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/paefremova/.ssh/id_rsa):
Enter passphrase for "/home/paefremova/.ssh/id_rsa" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/paefremova/.ssh/id_rsa
Your public key has been saved in /home/paefremova/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:[CLQnQJahc5KRDzz1BesU4FX1FHuEGPETYg/USF/nY paefremova@vbox
The key's randomart image is:
+---[RSA 4096]----+
|Xo .+ +oo.+*0*+ |
|o* B.o + o..=*
|o.*.= o . +.o.
| . . = + . +
| . . + o S . .
| . . o E|
| . . .
| . .
+---[SHA256]-----+
[paefremova@vbox ~]$
```

Рис. 4.3: Алгоритм 4096

- Создаю ключ по алгоритму ed25519 (рис. 4.4).

```

our public key has been saved in /home/paefremova/.ssh/id_rsa.pub
the key fingerprint is:
SHA256:iCLQnQJAYhc5KRdzz1BesU4FX1FHuEGPETYg/USF/nY paefremova@vbox
the key's randomart image is:
----[RSA 4096]----+
|o . + oo.+*+ |
|o* B.o + o..=* |
|o.*.= o . +o. |
|. . = + . + |
|. . + o S . |
|. . o E|
|. . |
|
----[SHA256]----+
paefremova@vbox ~]$ cd .ssh/
paefremova@vbox .ssh]$ ls
d_rsa id_rsa.pub known_hosts
paefremova@vbox .ssh]$ cat id_rsa.pub
-----BEGIN RSA PUBLIC KEY-----
MIIBIjANBQKAYhg87m15h6ij/+UX1fuqVrW0VnmQdF4UtuEjjajSBQg0r
i5fpCD6nbJDHY3vqBSGxewvHFxa1sb77A70tRoZ/J5UIF6j5CgArIgtHJpgxuqex3a6VM5rKL+x19znZbbtO+AUUzgfP0pqqV
i'pODlBhRqFxxJxx81oWxME1XpgwqrdrZ/x47F5Xcb215Ni6w1J9c+fAOVoKdRke1s19iInGcbu4fZuZHCoWLz9u0wM48c6BZxpa6
iqidfsV0f0f7b3/K2x0gMccgZoJMXLytdIptk9fJvD03H3zklU02jiYpppeYDlPvuUfxLMVzSRPe/Ey1wB802zSRGbwpALN1lTb0
'kw9YH1P4ZduwJKdh+MhiclaKnzVb5QyL7T7l0PuL51a5X8WThjKriyiEO+1QcL+w3SNPr4KNy3az7+gRwqR/gR4t94Ny285Yx+Kac
IH00D4NHFxGpTJ+Wj0oDua+qSAddyZfjGOBBtzMmhiit3kq2da90CXUTfuXg9LkV9Qn4sqAMNyBTL214rLi/6j4VtyrCJjcpqGSA
iZ77U8xEigX/ynfQ/hb4IwU6YqrE+fshEVpQ2v05ay+Tk4+pDysQ3lpy1fPa5JljlOP1L77JpeEuqDM/hKIP1UFkrigrYd6P9hoFV
/MOFQBQ== paefremova@vbox
-----END RSA PUBLIC KEY-----
paefremova@vbox .ssh]$ 

```

Рис. 4.4: Алгоритм ed25519

3. Добавляю скопированный SSH ключ на GitHub (рис. 4.5).

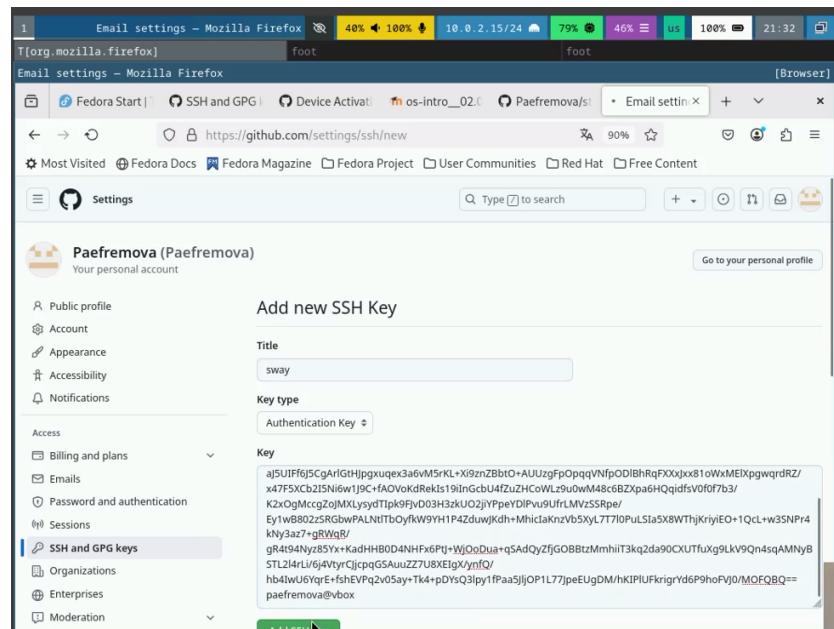


Рис. 4.5: Загрузка ключа на Github

4.4 Создание gpg ключа

- Генерирую gpg ключ (рис. 4.6).

The screenshot shows a terminal window with the following text:

```
[paefremova@vbox .ssh]$ gpg --full-generate-key
gpg (GnuPG) 2.4.5; Copyright (C) 2024 g10 Code GmbH
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

gpg: создан каталог '/home/paefremova/.gnupg'
Выберите тип ключа:
  (1) RSA and RSA
  (2) DSA and Elgamal
  (3) DSA (sign only)
  (4) RSA (sign only)
  (9) ECC (sign and encrypt) *default*
  (10) ECC (только для подписи)
  (14) Existing key from card
Ваш выбор? 1
длина ключей RSA может быть от 1024 до 4096.
Какой размер ключа Вам необходим? (3072) 4096
```

Рис. 4.6: GPG ключ

- Настраиваю этот gpg ключ указанным способом (рис. 4.7).

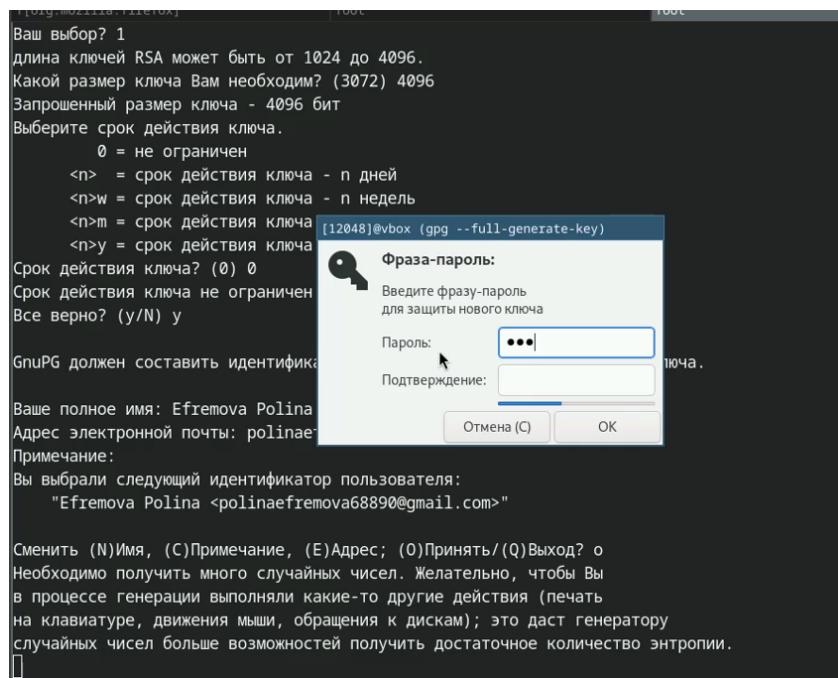


Рис. 4.7: GPG ключ - настройка

3. Вывожу список ключей и копирую отпечаток приватного ключа (рис. 4.8).

```
4@q\w b[paefremova@vbox .gnupg]$ cd ..
paefremova@vbox ~]$ gpg --list-secret-keys --keyid-format LONG
keyboxd]
-----
ec  rsa4096/B236D7B3A191ED01 2025-03-03 [SC]
      C3128B4B97409016F22BB885B236D7B3A191ED01
id      [ абсолютно ] Efremova Polina <polinaefremova68890@gmail.com>
sb  rsa4096/FBFEBAB78C50FCC02 2025-03-03 [E]

paefremova@vbox ~]$ gpg --armor --export B236D7B3A191ED01
```

Рис. 4.8: Список ключей

4. Копирую сгенерированный PGP ключ (рис. 4.9).

```
[keyboxd]
-----
sec rsa4096/B236D7B3A191ED01 2025-03-03 [SC]
    C3128B4B97409016FF22BB85B236D7B3A191ED01
uid          [ абсолютно ] Efremova Polina <polinaefremova68890@gmail.com>
ssb rsa4096/FBFEBEA78C50FCC02 2025-03-03 [E]

[paefremova@vbox ~]$ gpg --armor --export B236D7B3A191ED01
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQINBGff9foBEADe0e6wE+elrYrw/H09oMB00fmh0ZoB+pJYLU0Iz9/2ABZZp+K
yRaULFrT7QxsKEsi7BsnseekP7eEU95uL+9ubmQNQLz2W1f5xbob5VBLEEXC4z
7Ub7gnuY8MSmUnsKlCeJG8CvzRXopKXmLgMx0jmDQbaM2ZL4J1Nw+ChB9DBmcKf9
RTfa0x19FrjiXGf8wGIaJBf7JaoG5/Biu3Vv0antxI6yKyind2HRM50l7CQjN
yPw6f9XLSytdoZUTkdstYMUej5c7XymrlKCceOrLMoopPyunP4jlve9rjXZMFR6u
o1uj9ZIdYlwsenMbyIRWiAsnBqYwfXcYRRij2A3/riJERf4UUXP5tINQ0X3yS6PK
JPxrSGDhGcvKM+a3clmCQLEGHd0zwdhAtCob+5FnvC97lz15ZMksMdkbuvJKeo
lyUiirAWMLi6roR/zC0/zfqMPriGu810iA7rD6bm1FF/I9xnZpbqqj+W9ykJ60X9
B6uzyJClU3r2fbuD6L20+gzDWXm+A09gtqbenKUJc3pESRcsuVIB40jGLVuMApEW
E0oQ1kw89n9DA+kIDG8lwsliu7lGoddk/aen6D69KQRMAKEk0Upjbhm9bpkMrLh
E7szNM68kz1Et3y4t4H95d6p6hLKQ6c5SN11+KFRyNtYGIj5alGdsTi2wARAQAB
tC9FZnJlbW92YSBb2pbmEgPHBvbGluyWVmcmVtb3ZhNjg40TBAAZ21haWWuY29t
PokCUQQTaqgAOxYhBMMSi0uXQJAW/yk7hbI2170hke0BBQJnx6AhsDBQsJCAC
AiICBhUKCQgLAGQAgMBAh4HAheAAAoJELI2170hke0BGJYQAJMokjK2tu5srJLk
6mJMatAiZSUFD9G0x18fkwnyW+etS2jd8u/C67ZD8RCxIVcUqcDtS03s87dBhKz
Kot92NvxUk1TsVuadQMhwZd3m/mwcoaHpc/intDb9wpYLpoG0JAYltTzxWjN4aww
oQE5xCUr7Bu5JnktkJmXQnA6EaMjp/ua076vFle5mNHRB7Pe2yIAS1xyEIRWZ6L
```

Рис. 4.9: GPG ключ - копирование

5. Добавляю ключ на Github (рис. 4.10).

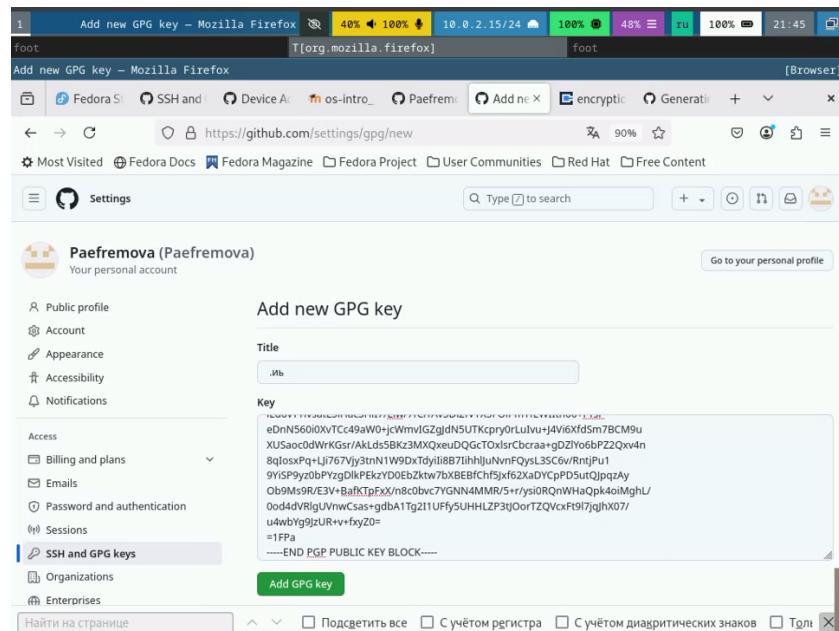


Рис. 4.10: GPG ключ - GitHub

6. Авторизуюсь, отвечаю на контрольные вопросы (рис. 4.11).

```
sec  rsa4096/B236D7B3A191ED01 2025-03-03 [SC]
      C3128B4B97409016FF22BB885B236D7B3A191ED01
uid          [ абсолютно ] Efremova Polina <polinaefremova68890@gmail.com>
ssb  rsa4096/FBFEBAB78C50FCC02 2025-03-03 [E]

[paefremova@vbox ~]$ git config --global user.signingkey B236D7B3A191ED01
[paefremova@vbox ~]$ git config --global commit.gpgsign true
[paefremova@vbox ~]$ git config --global gpg.program $(which gpg2)
[paefremova@vbox ~]$ gh auth login
? Where do you use GitHub? 1 [Use arrows to move, type to filter]
[paefremova@vbox ~]$ gh auth login
? Where do you use GitHub? GitHub.com
? What is your preferred protocol for Git operations on this host? SSH
? Upload your SSH public key to your GitHub account? /home/paefremova/.ssh/id_rsa.pub
? Title for your SSH key: zalupa konya
? How would you like to authenticate GitHub CLI? [Use arrows to move, type to filter]
> Login with a web browser
Paste an authentication token
```

Рис. 4.11: Авторизация

7. Ввожу код (рис. 4.12)

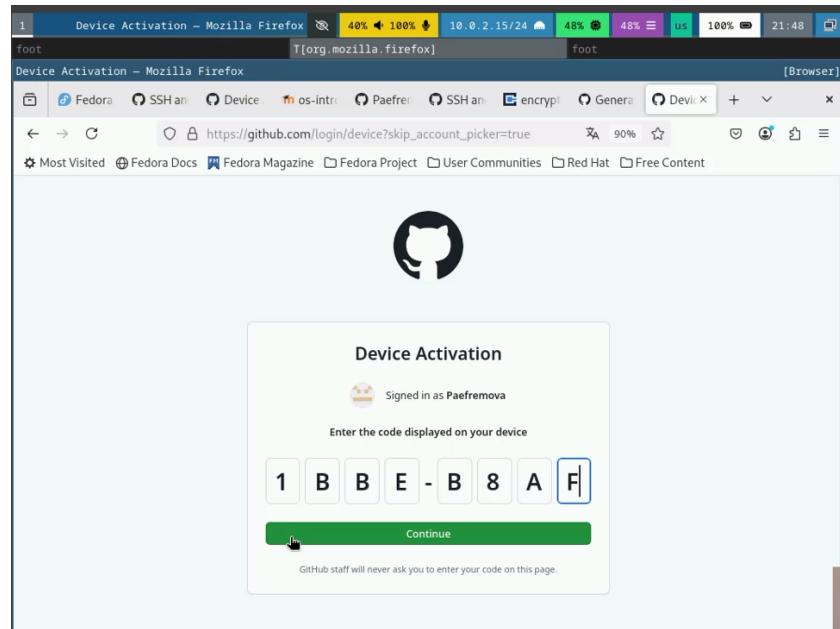


Рис. 4.12: Код

4.5 Создание репозитория курса на основе шаблона

1. Создаю репозиторий (рис. 4.13).

```
[root@vbox ~]# mkdir -p ~/work/study/2024-2025/"Операционные системы"
```

Рис. 4.13: Репозиторий

2. Перехожу в него и копирую шаблон (рис. 4.14).

```
se-directory-student-template --public
✓ Created repository Paefremova/study_2022-2023_os-intro on GitHub
  https://github.com/Paefremova/study_2022[root@vbox Операционные системы]# cd
[root@vbox ~]# mkdir -p ~/work/study/2024-2025/"Операционные системы"
[root@vbox ~]# cd ~/work/study/2024-2025[root@vbox Операционные системы]# gh repo create study_2024-2025_os-intro --template=ymadhamta/course-di
[root@vbox Операционные системы]# git clone --recursive git@github.com:<owner>/study_2024-2025_os-intro.git os-intro
-bash: owner: Нет такого файла или каталога

[root@vbox Операционные системы]# git clone --recursive git@github.com:paefremova/study_2024-2025_os-intro.git os-intro
fatal: bad boolean config value 'flase' for 'core.quotepath'
[root@vbox Операционные системы]# git clone --recursive git@github.com:Paefremova/study_2024-2025_os-intro.git os-intro
fatal: bad boolean config value 'flase' for 'core.quotepath'
[root@vbox Операционные системы]# cd
[root@vbox ~]# git clone --recursive git@github.com:Paefremova/study_2024-2025_os-intro.git os-intro
fatal: bad boolean config value 'flase' for 'core.quotepath'
[root@vbox ~]# git config --global core.quotepath false
[root@vbox ~]# git clone --recursive git@github.com:Paefremova/study_2024-2025_os-intro.git os-intro
Клонирование в «os-intro»...
The authenticity of host 'github.com (140.82.121.4)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvvV6TuJJhbpb2isF/zLDA0zPMsvHdkr4UvC0qU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? █
```

Рис. 4.14: Добавление шаблона

3. Перехожу в каталог курса (рис. 4.15).

```

! You were already logged in to this account
[raefremova@vbox ~]$ zbs
bash: zbs: команда не найдена
[raefremova@vbox ~]$ ls
config          pandoc-crossref      Видео        Общедоступные
Downloads       pandoc-crossref.1    Документы   'Рабочий стол'
pandoc-3.6.2    pandoc-crossref-Linux-X64.tar.xz Изображения Шаблоны
pandoc-3.6.2-linux-amd64.tar.gz work          Музыка

[raefremova@vbox ~]$ cd work/study/2024-2025/Операционные\ системы/
[raefremova@vbox Операционные системы]$ sudo rm -r os-intro/
[sudo] пароль для рафремова:
[raefremova@vbox Операционные системы]$ ls
[raefremova@vbox Операционные системы]$ git clone --recursive git@github.com:Paefremova/study_2024-2025_os-intro.git os-intro
Клонирование в «os-intro»...
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)
Получение объектов: 100% (36/36), 19.37 КиБ | 296.00 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/raefremova/work/study/2024-2025/Операционные системы/os-intro/template/presentation»...

```

Рис. 4.15: Каталог курса

4. Создаю необходимые каталоги (рис. 4.16).

```

computer-practice Компьютерный практикум по статистическому анализу данных
mathsec      Математические основы защиты информации и информационной безопасности
mathmod      Математическое моделирование
simulation-networks Моделирование сетей передачи данных
sciprog     Научное программирование
os-intro     Операционные системы
os2          Основы администрирования операционных систем
infosec-intro Основы информационной безопасности
[raefremova@vbox os-intro]$ make submodule
git submodule update --init --recursive
git submodule foreach 'git fetch origin; git checkout $(git rev-parse --abbrev-ref HEAD); git reset --hard origin/$(git rev-parse --abbrev-ref HEAD); git submodule update --recursive; git clean -dfx'
Entering 'template/presentation'
Указатель HEAD сейчас на коммите c9b2712 Merge branch 'release/1.0.4'
Entering 'template/report'
Указатель HEAD сейчас на коммите c26e22e Merge branch 'release/1.0.5'
[raefremova@vbox os-intro]$ nano COURSE
[raefremova@vbox os-intro]$ make prepare
[raefremova@vbox os-intro]$ ls
CHANGELOG.md  COURSE  LICENSE  package.json  presentation  README.en.md  README.md
config        labs    Makefile  prepare      project-personal  README.git-flow.md  template
[raefremova@vbox os-intro]$ git add .
[raefremova@vbox os-intro]$ ls
CHANGELOG.md  COURSE  LICENSE  package.json  presentation  README.en.md  README.md
config        labs    Makefile  prepare      project-personal  README.git-flow.md  template
[raefremova@vbox os-intro]$ rm package.json
[raefremova@vbox os-intro]$ git add .
[raefremova@vbox os-intro]$ git commit -am 'feat(main): make course structure'

```

Рис. 4.16: Необходимые каталоги

5. Отправляю файлы на сервер (рис. 4.17).

```
[paefremova@vbox os-intro]$ git push
Перечисление объектов: 40, готово.
Подсчет объектов: 100% (40/40), готово.
Сжатие объектов: 100% (30/30), готово.
Запись объектов: 100% (38/38), 342.30 КиБ | 2.17 МиБ/с, готово.
Total 38 (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:Paeefremova/study_2024-2025_os-intro.git
  f89fd86..ef3dd1e master -> master
```

Рис. 4.17: Сервер - загрузка

5 Выводы

В заключение хочется отметить, что данная лабораторная работа позволила мне научиться работать с системой Git. Я практиковала свои навыки в работе с командной строкой, теперь уже связывая выполнимое с директориями GitHub.

Кроме этого, я научилась работать с Markdown (вспомнила, опираясь на материалы из прошлого семестра в том числе)

6 Список литературы

1. Архитектура ЭВМ
2. Ефремова Полина, Лабораторная работа №2, Архитектура компьютеров
3. Лабораторная 2
4. лабораторная 3