

# **Отчёт по лабораторной работе №2**

**Дисциплина: архитектура компьютера**

**Закиров Нурислам Дамирович**

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Задание</b>	<b>5</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>6</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>8</b>
4.1	Настройка GitHub . . . . .	8
4.2	Базовая настройка Git . . . . .	9
4.3	Создание SSH-ключа . . . . .	11
4.4	Создание рабочего пространства и репозитория курса на основе шаблона . . . . .	13
4.5	Создание репозитория курса на основе шаблона . . . . .	13
4.6	Настройка каталога курса . . . . .	18
4.7	Выполнение заданий для самостоятельной работы . . . . .	20
<b>5</b>	<b>Выводы</b>	<b>24</b>

## Список иллюстраций

4.1	Предварительная конфигурация git . . . . .	10
4.2	Настройка кодировки . . . . .	10
4.3	Создание имени для начальной ветки . . . . .	10
4.4	Параметр autocrlf . . . . .	10
4.5	Параметр safecrlf . . . . .	10
4.6	Удаление файлов . . . . .	18
4.7	Создание каталогов . . . . .	18
4.8	Страница каталога в репозитории . . . . .	23

# 1 Цель работы

Цель этой работы состоит в том, чтобы изучить идеологию и использование инструментов контроля версий, а также получить практические навыки работы с системой git.

## 2 Задание

1. Настройка GitHub.
2. Базовая настройка Git.
3. Создание SSH-ключа.
4. Создание рабочего пространства и репозитория курса на основе шаблона.
5. Создание репозитория курса на основе шаблона.
6. Настройка каталога курса.
7. Выполнение заданий для самостоятельной работы.

### 3 Теоретическое введение

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 изменения, сделанные разными участниками, вручную выбрать нужную версию, отменить изменения вовсе или заблокировать файлы для изменения. В зависимости от настроек блокировка не позволяет

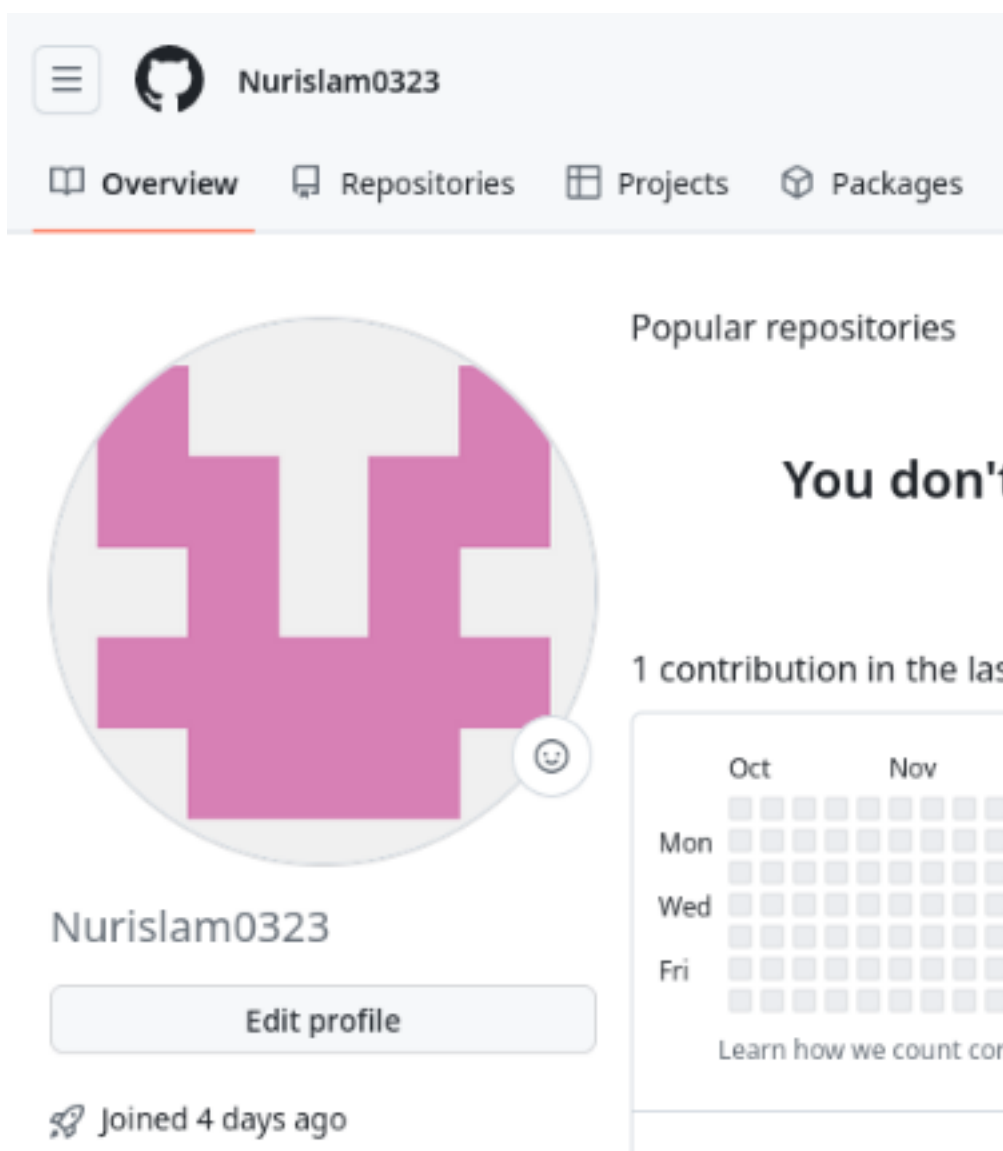
другим пользователям получить рабочую копию или препятствует изменению рабочей копии файла средствами файловой системы ОС, обеспечивая таким образом привилегированный доступ только одному пользователю, работающему с файлом. Системы контроля версий также могут обеспечивать дополнительные, более гибкие 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. 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. Работа пользователя со своей веткой начинается с проверки и получения изменений из центрального репозитория (при этом в локальное дерево до начала этой процедуры не должно было вноситься изменений). Затем можно вносить изменения в локальное дерево и/или ветке. После завершения внесения какого-то изменения в файлы и/или каталоги проекта необходимо разместить их в центральном репозитории.

## **4 Выполнение лабораторной работы**

### **4.1 Настройка GitHub**

Создаю учетную запись на сайте GitHub. Аккаунт создан (рис. [??]).





){#fig:001

width=70%}

## 4.2 Базовая настройка Git

Запускаем терминал, после чего делаю предварительную конфигурацию git. Ввожу команду `git config --global user.name`, указывая свое имя и команду `git config --global user.email "work@mail"`, указывая в ней мою электронную почту (рис. [4.1]).

```
[nurislamzakirov@fedora ~]$ git config --global user.name "<Nurislam Zakirov>"
[nurislamzakirov@fedora ~]$ git config --global user.email "<nuric35737@gmail.com>"
[nurislamzakirov@fedora ~]$
```

Рис. 4.1: Предварительная конфигурация git

Настраиваю git для корректного отображения символов (рис. [4.2]).

```
[nurislamzakirov@fedora ~]$ git config --global core.quotepath false
[nurislamzakirov@fedora ~]$
```

Рис. 4.2: Настройка кодировки

Задаю имя «master» для начальной ветки (рис. [4.3]).

```
[nurislamzakirov@fedora ~]$ git config --global init.defaultBranch master
[nurislamzakirov@fedora ~]$
```

Рис. 4.3: Создание имени для начальной ветки

Даём параметр autocrlf со значением input (рис. [4.4]).

```
[nurislamzakirov@fedora ~]$ git config --global core.autocrlf input
[nurislamzakirov@fedora ~]$
```

Рис. 4.4: Параметр autocrlf

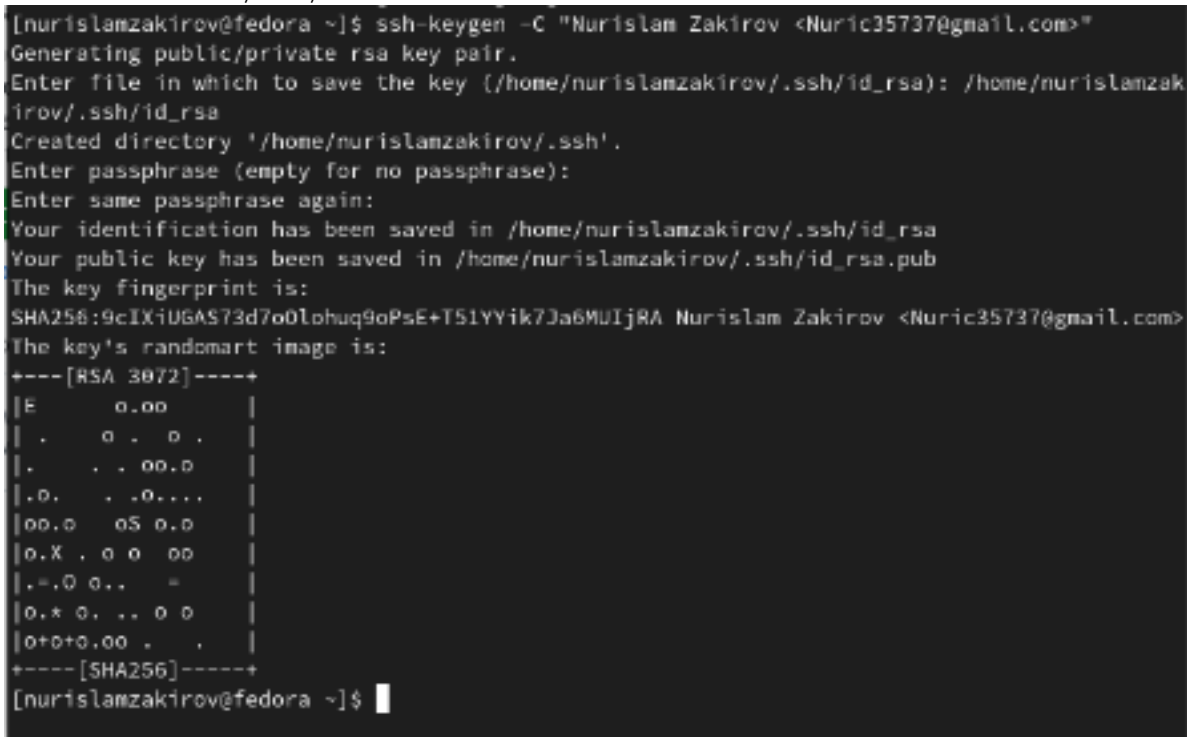
Задаю параметр safecrlf со значением warn, так Git будет проверять преобразование на обратимость (рис. [4.5]). При значении warn Git только выведет предупреждение, но будет принимать необратимые конвертации.

```
[nurislamzakirov@fedora ~]$ git config --global core.safecrlf warn
[nurislamzakirov@fedora ~]$
```

Рис. 4.5: Параметр safecrlf

## 4.3 Создание SSH-ключа

Ввожу команду `ssh-keygen -C "Имя Фамилия, work@email"`, указывая имя владельца и электронную почту владельца (рис. [??]). Ключ автоматически сохранится в каталоге `~/.ssh/`.

A terminal window showing the execution of the `ssh-keygen` command. The user is prompted to enter a comment, a file name to save the key, a passphrase, and a confirmation of the passphrase. The output shows the key pair generated, the file paths for the private and public keys, the key fingerprint, and a randomart image for the key.

```
[nurislamzakirov@fedora ~]$ ssh-keygen -C "Nurislam Zakirov <Nuric35737@gmail.com>"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/nurislamzakirov/.ssh/id_rsa): /home/nurislamzakirov/.ssh/id_rsa
Created directory '/home/nurislamzakirov/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/nurislamzakirov/.ssh/id_rsa
Your public key has been saved in /home/nurislamzakirov/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:9cIXiUGAS73d7o0lohuq9oPsE+T51YYik7Ja6MUljRA Nurislam Zakirov <Nuric35737@gmail.com>
The key's randomart image is:
+---[RSA 3072]-----+
|E      o.o      |
| .    o . o .   |
| .    . . oo.o   |
|.o.    . .o....  |
|oo.o    oS o.o   |
|o.X . o o oo    |
| .o.o o.. =     |
|o.* o. . . o o  |
|o+o+o.o . .    |
+---[SHA256]-----+
[nurislamzakirov@fedora ~]$
```

#fig:007 width=70% }

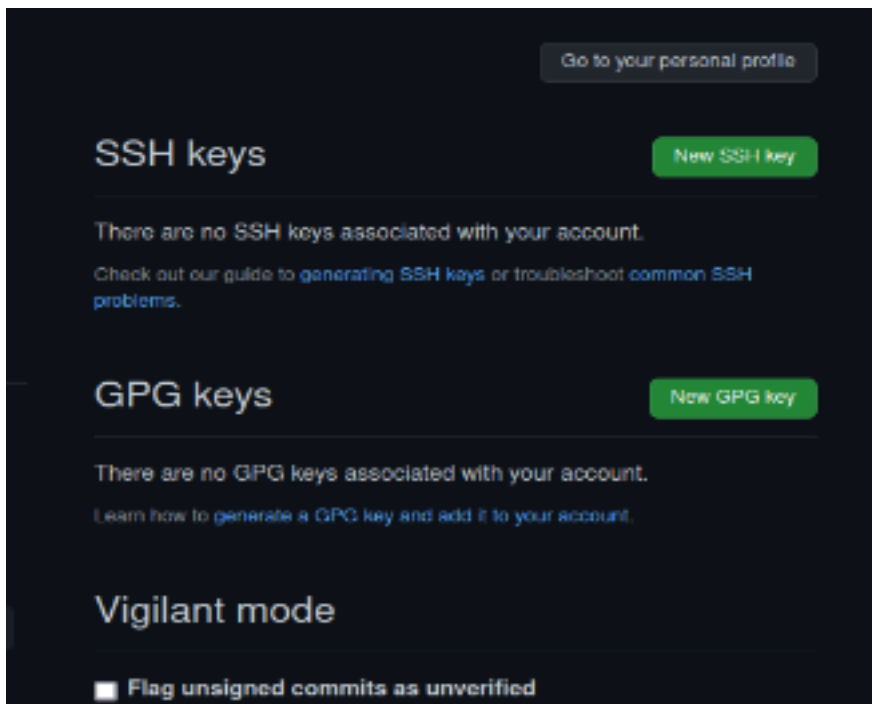
Копирую открытый ключ из директории, в которой он был сохранен, с помощью утилиты `xclip` (рис. [??]).

A terminal window showing the command `cat ~/.ssh/id_rsa.pub | xclip -sel clip~` being executed to copy the public key to the clipboard.

```
[nurislamzakirov@fedora ~]$ cat ~/.ssh/id_rsa.pub | xclip -sel clip~
```

#fig:008 width=70% }

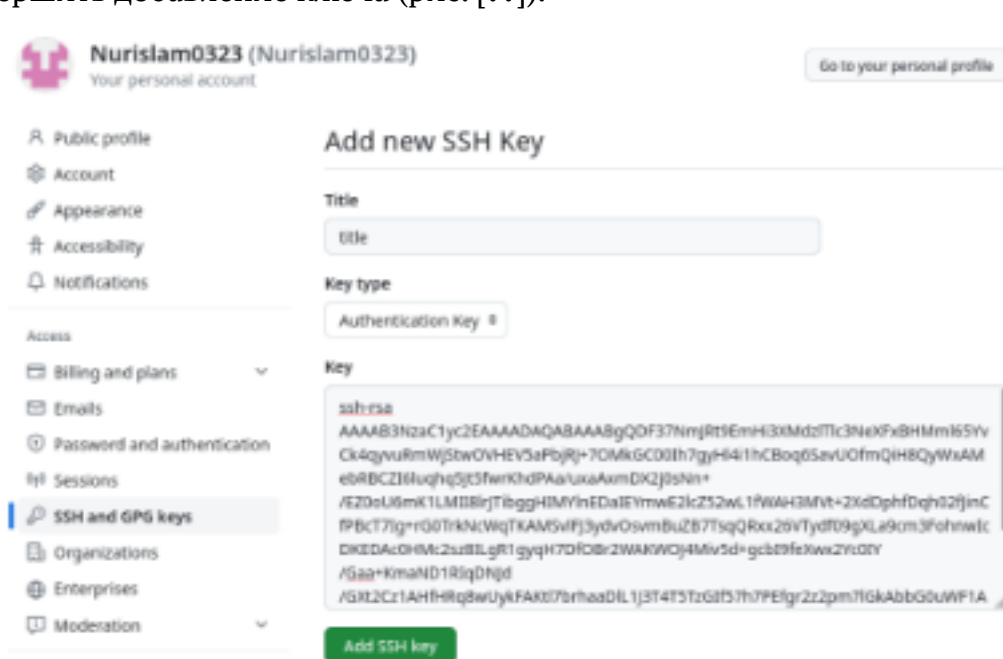
Открываю браузер, захожу на сайт GitHub. Открываю свой профиль и выбираю страницу «SSH and GPG keys». Нажимаю кнопку «New SSH key» (рис. [??]).



} {#fig:009 width=70%

}

Вставляю скопированный ключ в поле «Key». Нажимаю «Add SSH-key», чтобы завершить добавление ключа (рис. [??]).

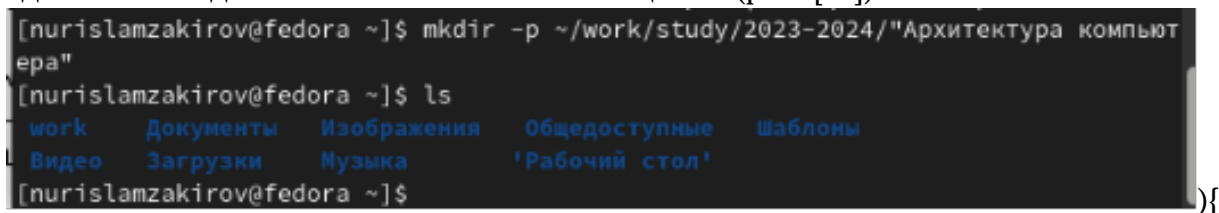


} {

#fig:010 width=70% }

## 4.4 Создание рабочего пространства и репозитория курса на основе шаблона

Я закрываю браузер и запускаю терминал. Создаю директорию и рабочее пространство с помощью утилиты `mkdir`, используя ключ `-p` после домашней директории `~/work/study/2022-2023/Computer Architecture`. Далее я проверяю, были ли созданы необходимые мне каталоги с помощью `ls` (рис. [??]).

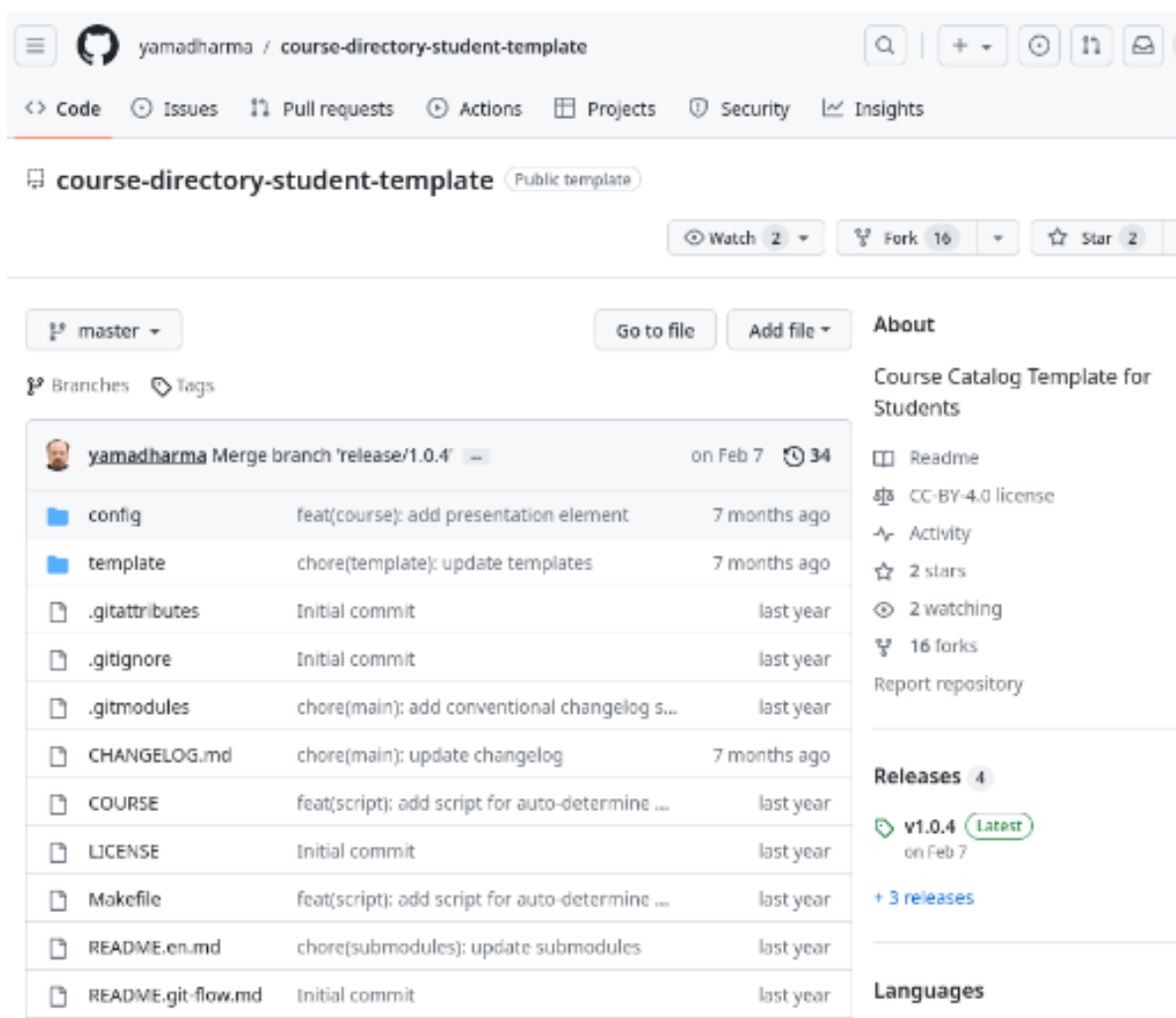


```
[nurislamzakirov@fedora ~]$ mkdir -p ~/work/study/2023-2024/"Архитектура компьютера"
[nurislamzakirov@fedora ~]$ ls
work      Документы  Изображения  Общедоступные  Шаблоны
Видео     Загрузки   Музыка       'Рабочий стол'
```

#fig:011 width=70% }

## 4.5 Создание репозитория курса на основе шаблона

Заходим на страницу репозитория с шаблоном курса по адресу <https://github.com/yamadharma/directory-student-template>. Выбираем «Use this template», чтобы использовать этот шаблон для своего репозитория (рис. [??]).



#fig:012 width=70% }

В открывшемся окне задаю имя репозитория (Repository name): study\_2023–2024\_arh-  
ps и создаю его, нажав кнопку «Create repository from template» (рис. [??]).

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Required fields are marked with an asterisk (\*).

Owner \*

 Nurislam0323

Repository name \*

/ study\_2023-2024\_arh-pc

✔ study\_2023-2024\_arh-pc is available.

Great repository names are short and memorable. Need inspiration? How about [fuzzy-carnival](#) ?

Description (optional)



**Public**

Anyone on the internet can see this repository. You choose who can commit.



**Private**

You choose who can see and commit to this repository.



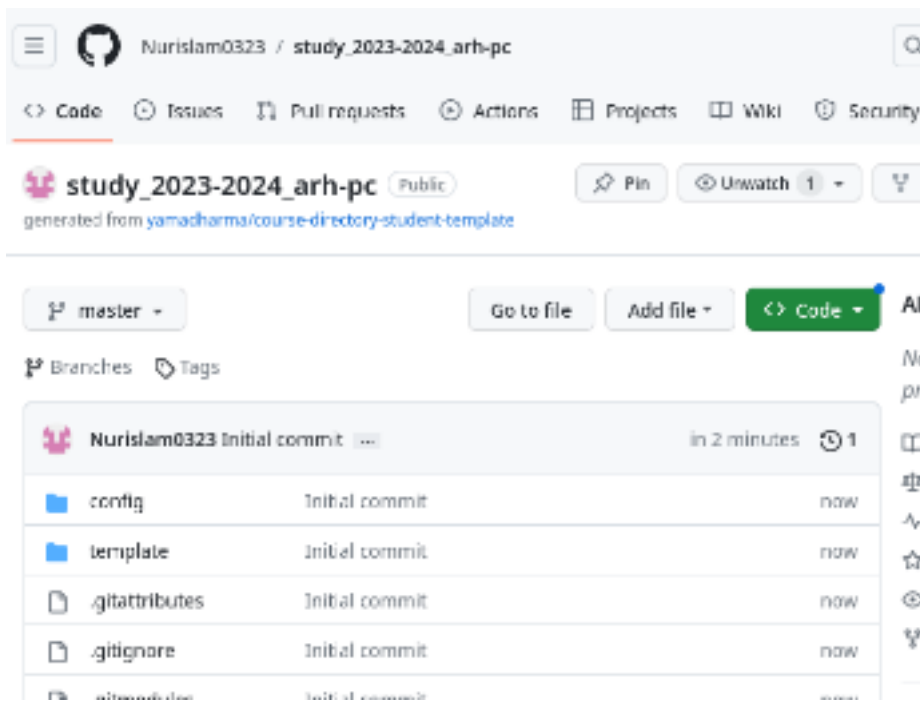
You are creating a public repository in your personal account.

Create repository

)

#fig:013 width=70% }

Репозиторий создан (рис. [??]).



#fig:014

width=70% }

Через терминал перехожу в созданный каталог курса с помощью утилиты cd (рис. [??]).

```
[nurislamzakirov@fedora ~]$ cd ~/work/study/2023-2024/'Архитектура компьютера'
[nurislamzakirov@fedora Архитектура компьютера]$
```

)

#fig:015 width=70% }

Клонирую созданный репозиторий с помощью команды `git clone -recursive git@github.com:/study_2022-2023_arh-pc.git arch-pc` (рис. [??]).



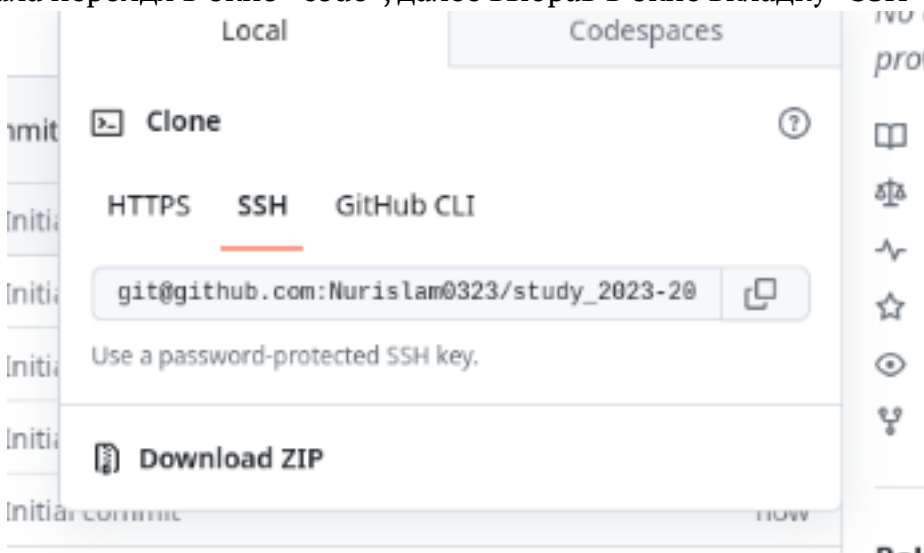
```

[nurislamzakirov@fedora Архитектура компьютера]$ git clone --recursive git@github.com:Nurislam0323/study_2023-2024_arch-pc.git arch-pc
Клонирование в «arch-pc»...
The authenticity of host 'github.com (148.82.121.4)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvV9tUJJhnpZisF/zLD48zPMSvHdKtr4UvCOqU.
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: 27, done.
remote: Counting objects: 100% (27/27), done.
remote: Compressing objects: 100% (26/26), done.
remote: Total 27 (delta 1), reused 11 (delta 6), pack-reused 6
Получение объектов: 100% (27/27), 16.93 КиБ | 289.86 КиБ/с, готово.
Определение изменений: 100% (1/1), готово.
Подмодуль «template/presentation» (https://github.com/yanadharna/academic-presentation-markdown-template.git) зарегистрирован по пути «template/presentation»
Подмодуль «template/report» (https://github.com/yanadharna/academic-laboratory-report-template.git) зарегистрирован по пути «template/report»
Клонирование в «/home/nurislamzakirov/work/study/2023-2024/Архитектура компьютера/arch-pc/template/presentation»...
remote: Enumerating objects: 82, done.
remote: Counting objects: 100% (82/82), done.
remote: Compressing objects: 100% (57/57), done.
remote: Total 82 (delta 28), reused 77 (delta 23), pack-reused 6
Получение объектов: 100% (82/82), 52.99 КиБ | 1.18 КиБ/с, готово.
Определение изменений: 100% (28/28), готово.
Клонирование в «/home/nurislamzakirov/work/study/2023-2024/Архитектура компьютера/arch-pc/template/report»...
remote: Enumerating objects: 161, done.
remote: Counting objects: 100% (161/161), done.
remote: Compressing objects: 100% (70/70), done.
remote: Total 161 (delta 40), reused 88 (delta 27), pack-reused 9
Получение объектов: 100% (161/161), 327.25 КиБ | 1.94 КиБ/с, готово.
Определение изменений: 100% (46/46), готово.
Submodule path 'template/presentation': checked out 'b1be3538ee91f58092b4cb75b6318174548b753e'
Submodule path 'template/report': checked out '1d1b61dcac9c287a83917b82e3ae11a33b1c3b2'
[nurislamzakirov@fedora Архитектура компьютера]$

```

#fig:016 width=70% }

Копирую ссылку для клонирования на странице созданного репозитория, сначала перейдя в окно «code», далее выбрав в окне вкладку «SSH» (рис. [??]).



width=70% }

#fig:017

## 4.6 Настройка каталога курса

Перехожу в каталог arch-pc с помощью утилиты cd (рис. [??]).

```
[nurislamzakirov@fedora Архитектура компьютера]$ cd ~/work/study/2023-2024/'Архитектура компьютера'  
[nurislamzakirov@fedora Архитектура компьютера]$ cd arch-pc  
[nurislamzakirov@fedora arch-pc]$
```

#fig:018 width=70% }

Удаляю лишние файлы с помощью утилиты rm (рис. [4.6]).

```
[nurislamzakirov@fedora arch-pc]$ rm package.json  
[nurislamzakirov@fedora arch-pc]$
```

Рис. 4.6: Удаление файлов

Создаю необходимые каталоги (рис. [4.7]).

```
[nurislamzakirov@fedora arch-pc]$ echo arch-pc > COURSE  
[nurislamzakirov@fedora arch-pc]$ make
```

Рис. 4.7: Создание каталогов

Добавляю все созданные каталоги с помощью git add, комментирую и сохраняю изменения на сервере как добавление курса с помощью git commit (рис. [??]).

```
[nurislamzakirov@fedora arch-pc]$ git add .
[nurislamzakirov@fedora arch-pc]$ git commit -am 'feat(main): make course structure'
[master 0e51393] feat(main): make course structure
199 files changed, 54725 insertions(+), 14 deletions(-)
create mode 100644 labs/README.md
create mode 100644 labs/README.ru.md
create mode 100644 labs/lab01/presentation/Makefile
create mode 100644 labs/lab01/presentation/image/kulyabov.jpg
create mode 100644 labs/lab01/presentation/presentation.md
create mode 100644 labs/lab01/report/Makefile
create mode 100644 labs/lab01/report/bib/cite.bib
create mode 100644 labs/lab01/report/image/placeimg_800_600_tech.jpg
create mode 100644 labs/lab01/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100755 labs/lab01/report/pandoc/filters/pandoc_eqnos.py
create mode 100755 labs/lab01/report/pandoc/filters/pandoc_fignos.py
create mode 100755 labs/lab01/report/pandoc/filters/pandoc_secnos.py
create mode 100755 labs/lab01/report/pandoc/filters/pandoc_tablenos.py
create mode 100644 labs/lab01/report/pandoc/filters/pandocxnos/__init__.py
create mode 100644 labs/lab01/report/pandoc/filters/pandocxnos/core.py
create mode 100644 labs/lab01/report/pandoc/filters/pandocxnos/main.py
create mode 100644 labs/lab01/report/pandoc/filters/pandocxnos/pandocattributes.py
create mode 100644 labs/lab01/report/report.md
create mode 100644 labs/lab02/presentation/Makefile
create mode 100644 labs/lab02/presentation/image/kulyabov.jpg
create mode 100644 labs/lab02/presentation/presentation.md
create mode 100644 labs/lab02/report/Makefile
create mode 100644 labs/lab02/report/bib/cite.bib
create mode 100644 labs/lab02/report/image/placeimg_800_600_tech.jpg
create mode 100644 labs/lab02/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100755 labs/lab02/report/pandoc/filters/pandoc_eqnos.py
create mode 100755 labs/lab02/report/pandoc/filters/pandoc_fignos.py
create mode 100755 labs/lab02/report/pandoc/filters/pandoc_secnos.py
```


#fig:021 width=70% }

Отправляю все на сервер с помощью push (рис. [??]).

```
[nurislamzakirov@fedora arch-pc]$ git push
Перечисление объектов: 37, готово.
Подсчет объектов: 100% (37/37), готово.
При сжатии изменений используется до 3 потоков
Сжатие объектов: 100% (29/29), готово.
Запись объектов: 100% (35/35), 342.14 КиБ | 2.61 МиБ/с, готово.
Всего 35 (изменений 4), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (4/4), completed with 1 local object.
To github.com:Nurislam0323/study_2023-2024_arh-pc.git
 56d5b2e..0e51393 master -> master
[nurislamzakirov@fedora arch-pc]$
```

#fig:022 width=70% }

Проверяю верность выполнения работы сначала на самом сайте GitHub (рис. [??]).

<div>  <b>Nurislam0323</b> feat(main): make course structure         </div> <div> 0e51393 · 1 minute ago <div>History</div> </div>		
Name	Last commit message	Last commit date
..		
lab01	feat(main): make course structure	1 minute ago
lab02	feat(main): make course structure	1 minute ago
lab03	feat(main): make course structure	1 minute ago
lab04	feat(main): make course structure	1 minute ago
lab05	feat(main): make course structure	1 minute ago
lab06	feat(main): make course structure	1 minute ago
lab07	feat(main): make course structure	1 minute ago
lab08	feat(main): make course structure	1 minute ago
lab09	feat(main): make course structure	1 minute ago
lab10	feat(main): make course structure	1 minute ago
lab11	feat(main): make course structure	1 minute ago
README.md	feat(main): make course structure	1 minute ago
README.ru.md	feat(main): make course structure	1 minute ago

#fig:023 width=70% }

## 4.7 Выполнение заданий для самостоятельной работы

1. Перехожу в директорию labs/lab02/report с помощью утилиты cd. Создаю в каталоге файл для отчета по второй лабораторной работе с помощью утилиты touch (рис. [??]).

```
[nurislamzakirov@fedora arch-pc]$ cd labs/lab02/report
[nurislamzakirov@fedora report]$ touch Л02_Закиров_Нурислам_отчет
[nurislamzakirov@fedora report]$
```

#fig:024 width=70% }

Перемещаю первую лабораторную работу в /home/nurislamzakirov/work/study/2023-2024/Архитектура компьютеров/arch-pc/labs/lab01/report , после чего перемещаюсь в терминале по этому же пути и добавляю данный файл в репозиторий с

помощью git add, а затем сохраняю изменения с помощью git commit, и в конце отправляю в центральный репозиторий сохраненные изменения командой git push -f origin master (рис. [??]).

```
[nurislamzakirov@fedora report]$ cd .
[nurislamzakirov@fedora report]$ cd ..
[nurislamzakirov@fedora lab02]$ cd ..
[nurislamzakirov@fedora labs]$ cd lab01
[nurislamzakirov@fedora lab01]$ cd report
[nurislamzakirov@fedora report]$ git add /01_Закиров_Нурислам.docx
[nurislamzakirov@fedora report]$ git add /01_Закиров_Нурислам.docx
[nurislamzakirov@fedora report]$ git commit -m "Add existing file"
[master 26b0f67] Add existing file
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 labs/lab01/report/01_Закиров_Нурислам.docx
[nurislamzakirov@fedora report]$ git push -f origin master
Перечисление объектов: 10, готово.
Подсчет объектов: 100% (10/10), готово.
При сжатии изменений используется до 3 потоков
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 2.85 МиБ | 1.97 МиБ/с, готово.
Всего 6 (изменений 3), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To github.com:Nurislam0323/study_2023-2024_arh-pc.git
0e51393..26b0f67 master -> master
[nurislamzakirov@fedora report]$
```

#fig:025 width=70% }

Проверяю на сайте GitHub верность пройденных действий. Мы видим, что теперь в нашем репозитории добавились отчеты по выполнению первой и второй лабораторной работе (рис. [??], рис. [4.8]).

Nurislam0323 / study\_2023-2024\_arh-pc

+

<> Code

Issues

Pull requests

Actions

Projects

Wiki

Security

Insights

...

master

study\_2023-2024\_arh-pc / labs / lab01

Go to file

Add file

...

/ report /

Nurislam0323

Add existing file

26b0f67 · 1 minute ago

History

Name	Last commit message	Last commit date
..		
bit	feat(main): make course structure	22 minutes ago
image	feat(main): make course structure	22 minutes ago
pandoc	feat(main): make course structure	22 minutes ago
Makefile	feat(main): make course structure	22 minutes ago
report.md	feat(main): make course structure	22 minutes ago
ЛО1_Закиров_Нурислам.docx	Add existing file	1 minute ago

#fig:026 width=70% }

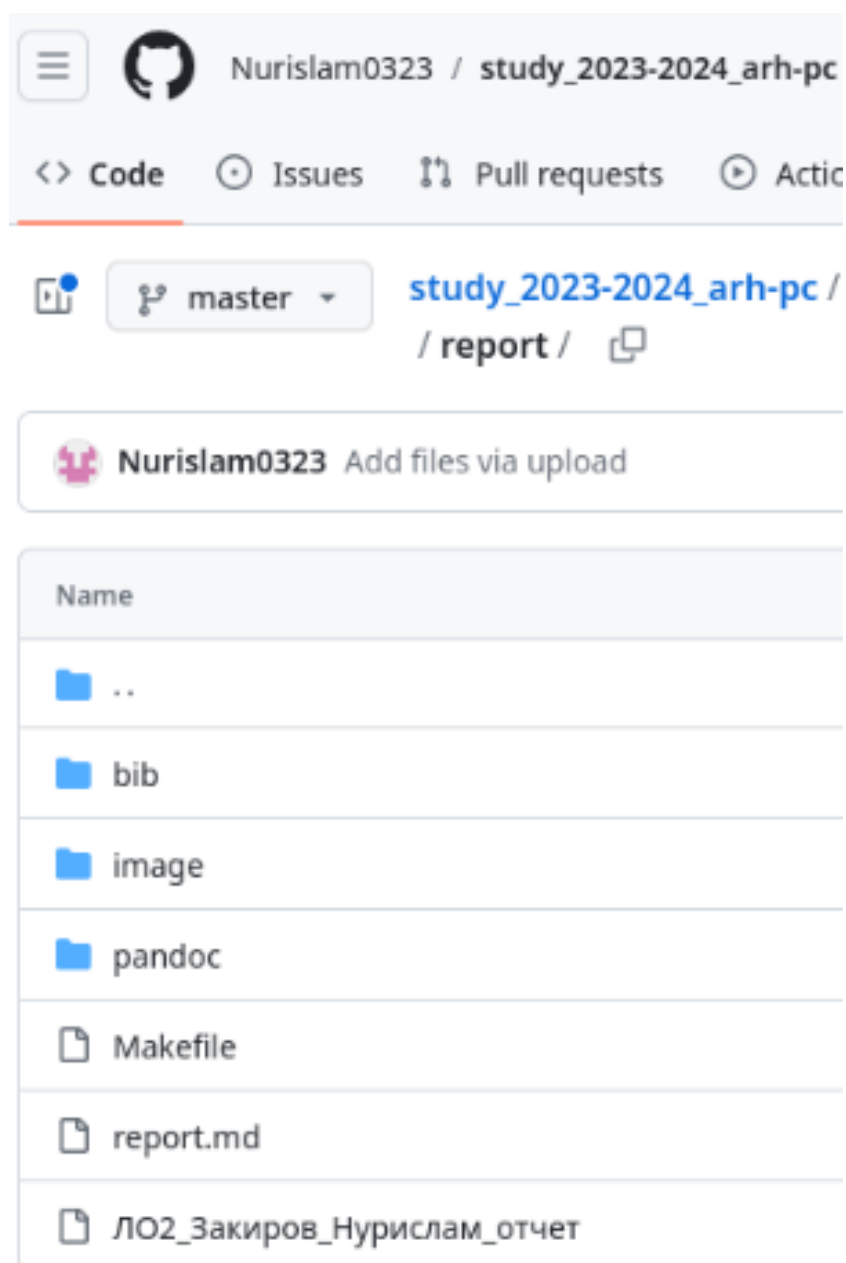


Рис. 4.8: Страница каталога в репозитории

## 5 Выводы

При выполнении данной лабораторной работы я изучил идеологию и применение средств контроля версий, а также приобрел практические навыки по работе с системой git. # Список литературы

1. Архитектура ЭВМ
2. Git - gitattributes Документация