

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: Козина Д. А.

Группа: НММбд-02-24

МОСКВА

2024 г.

Содержание

1 Цель работы	3
2 Выполнение лабораторной работы.....	4
3 Выполнение заданий для самостоятельной работы.....	10
4 Выводы.....	11

1 Цель работы

Изучение идеологии и применения средств контроля версий. Приобретение практических навыков по работе с системой git.

2 Выполнение лабораторной работы

2.1 Настройка github

Создадим аккаунт на сайте github (рис. 2.1).

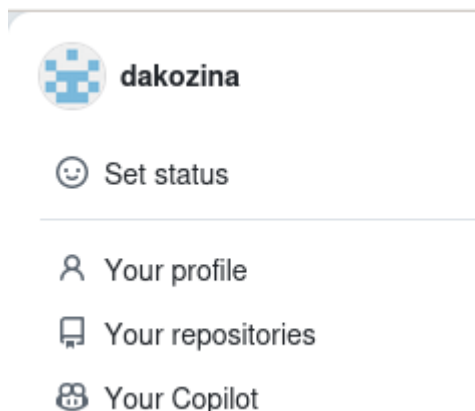


рис. 2.1: Аккаунт в github

2.2 Базовая настройка git

С помощью команды `git config --global` сделаем предварительную конфигурацию git, указав имя и email, настроим utf-8 в выводе сообщений git, зададим имя начальной ветки (master), параметры `autocrlf` и `safecrlf` (рис. 2.2).

```
dasha-lab@Dasha-lab:~$ git config --global user.name "<Daria Kozina>"
dasha-lab@Dasha-lab:~$ git config --global user.email "<1132246831@pfur.ru>"
dasha-lab@Dasha-lab:~$ git config --global core.quotepath false
dasha-lab@Dasha-lab:~$ git config --global init.defaultBranch master
dasha-lab@Dasha-lab:~$ git config --global core.autocrlf input
dasha-lab@Dasha-lab:~$ git config --global core.safecrlf warn
```

рис. 2.2: Использование команды `git config --global`

2.3 Создание SSH ключа

Сгенерируем пару ключей (приватный и открытый) для последующей идентификации пользователя на сайте репозиториях, с помощью команды `ssh-keygen -C` укажу свои имя и email. Ключ автоматически сохранится в каталоге `~/.ssh/`. (рис. 2.3).

```
dasha-lab@Dasha-lab:~$ ssh-keygen -C "Daria Kozina <1132246831@pfur.ru>"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/dasha-lab/.ssh/id_rsa):
Created directory '/home/dasha-lab/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/dasha-lab/.ssh/id_rsa
Your public key has been saved in /home/dasha-lab/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:tE2+aG3U/4pyMMXvmPKVAK8y0p8EiQk2WADsN0eJR90 Daria Kozina <1132246831@pfur.ru>
The key's randomart image is:
+---[RSA 3072]-----+
|  ....**o .      |
|  . o.+o.oE.     |
|  .  o  + +.     |
|  . o .. * oo     |
|  . o  S O.+     |
|              =oX o.. |
|              o +o=++ |
|  . .o ++..     |
|              =o ... |
+-----[SHA256]-----+
```

рис. 2.3: Генерация SSH-ключа

Заходим на сайт GitHub, открываем свой профиль, переходим в «Setting», затем «SSH and GPG keys» и нажимаем кнопку «New SSH key», тем самым загрузив сгенеренный открытый ключ. Копируем ключ из локальной консоли в буфер обмена (рис. 2.4), вставим ключ в появившееся на сайте поле и называем ключ «Title», нажав «Add SSH-key», завершаем добавление ключа (рис. 2.5).

```
dasha-lab@Dasha-lab:~$ cat ~/.ssh/id_rsa.pub | xclip -sel clip
dasha-lab@Dasha-lab:~$
```

рис. 2.4: Копирование содержимого

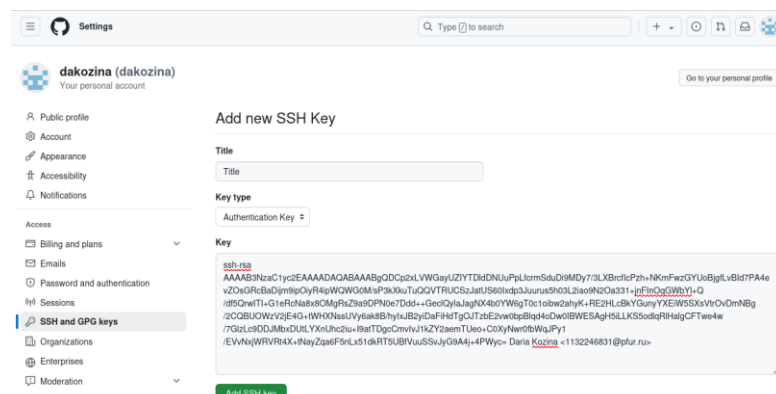


Рис. 2.5: Добавление ключа

2.4 Сознание рабочего пространства и репозитория курса на основе шаблона

С помощью команды `mkdir -p` создадим директорию после домашней папки

рекурсивно. Командой ls проверим создание каталогов (рис. 4.6).

```
dasha-lab@Dasha-lab:~$ mkdir -p work/study/2024-2025/"Архитектура компьютера"
dasha-lab@Dasha-lab:~$ ls
work      Документы  Изображения  Общедоступные  Шаблоны
Видео     Загрузки  Музыка       'Рабочий стол'
```

Рис. 2.6: Создание рабочего пространства

2.5 Создание репозитория курса на основе шаблона

Перейдем на страницу репозитория с шаблоном курса. Выбираем «Use this template» (рис. 2.7).

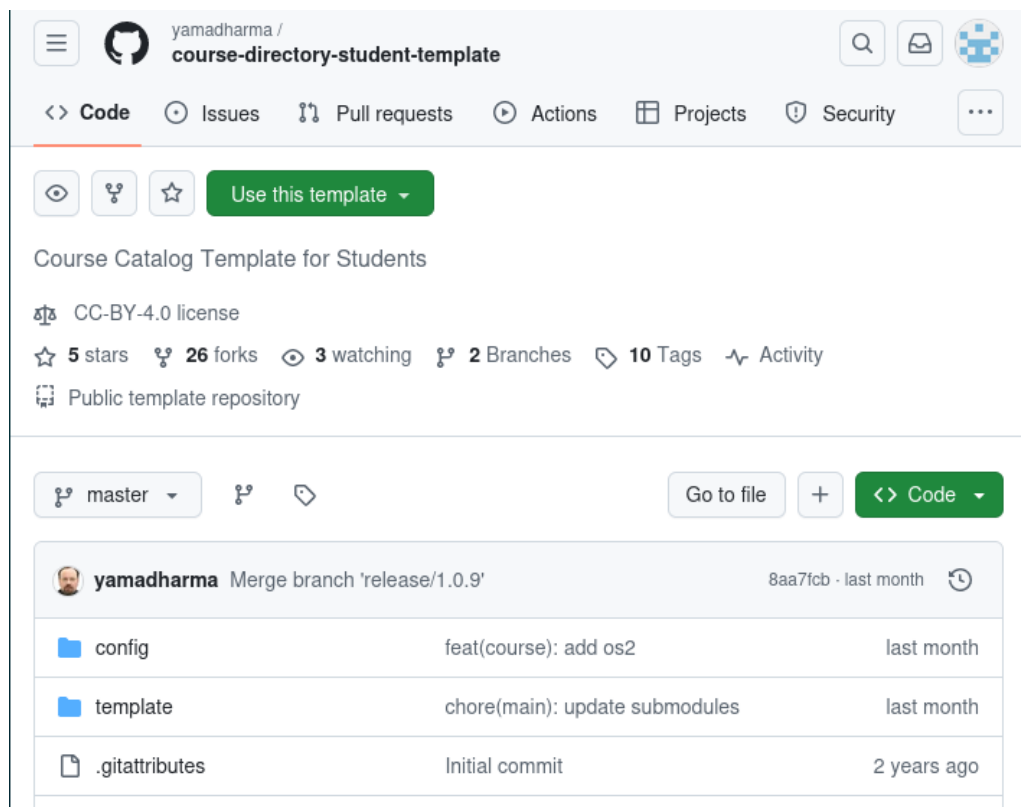


рис. 2.7: Страница шаблона для репозитория

В открывшемся окне задаем имя репозиторию и, нажав «Create repository», создаем репозиторий (рис. 2.8). Репозиторий создан (рис. 2.9).

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 (*).

Repository template

 yamadharma/course-directory-student-template ▾

Start your repository with a template repository's contents.

☐ Include all branches

Copy all branches from yamadharma/course-directory-student-template and not just the default branch.

Owner *

 dakozina ▾

Repository name *

study_2024-2025_arh-pc

✔ study_2024-2025_arh-pc is available.

Great repository names are short and memorable. Need inspiration? How about **upgraded-fishstick** ?

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



рис. 2.8: Окно создания репозитория

 **study_2024-2025_arh-pc** Public

generated from [yamadharma/course-directory-student-template](#)

 Pin  Unwatch 1

 master ▾  1 Branch  0 Tags

 Add file ▾  Code ▾








 dakozina	Initial commit	d0fc60f · now	 1 Commit
 config	Initial commit		now
 template	Initial commit		now
 .gitattributes	Initial commit		now
 .gitignore	Initial commit		now
 .gitmodules	Initial commit		now

рис. 2.9: Созданный репозиторий

С помощью команды `cd` перейдем в каталог курса, клонируем созданный репозиторий командой `git clone --recursive` (рис. 2.10). Ссылку для клонирования копируем на странице созданного репозитория, перейдя в окно «Code», далее вкладку «SSH» (рис. 2.11).

```
dasha-lab@Dasha-lab:~$ cd ~/work/study/2024-2025/"Архитектура компьютера"
dasha-lab@Dasha-lab:~/work/study/2024-2025/Архитектура компьютера$ git clone --recursive
git@github.com:dakozina/study_2024-2025_arh-pc.git arch-pc
Клонирование в «arch-pc»...
The authenticity of host 'github.com (140.82.121.3)' can't be established.
```

рис. 2.10: Клонирование репозитория

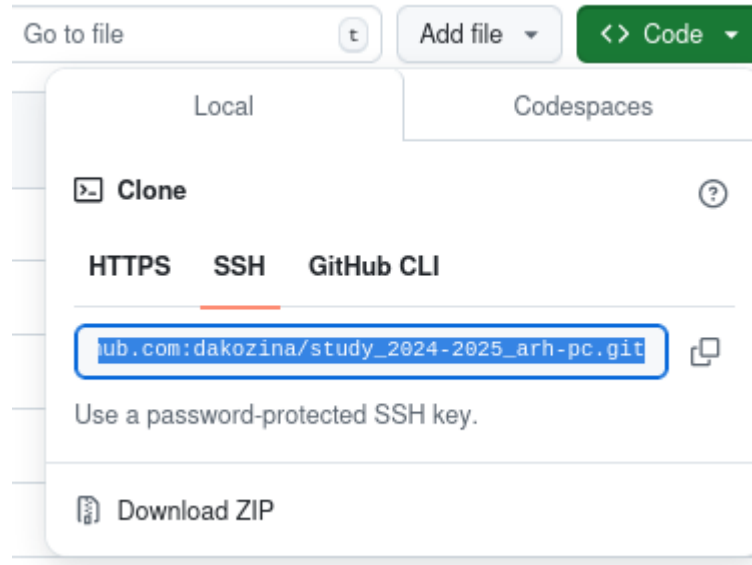


рис. 2.11: Ссылка для клонирования репозитория

2.6 Настройка каталога курса

С помощью команды `cd` перейдем в каталог курса, удалим лишние файлы командой `rm` (рис. 2.12).

```
dasha-lab@Dasha-lab:~$ cd ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc
dasha-lab@Dasha-lab:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ rm package.json
```

рис. 2.12: Перемещение между директориями, удаление файлов

Создадим необходимые каталоги (рис. 2.13).

```
dasha-lab@Dasha-lab:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ echo arch-pc > COURSE
dasha-lab@Dasha-lab:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ make
```

рис. 2.13: Создание каталогов

Отправим созданные каталоги на сервер. С помощью `git add` добавим все созданные каталоги, прокомментируем и сохраняем изменения на сервере как добавление курса с помощью `git commit`. С помощью `push` отправим все на сервер (рис. 2.14).


```

dasha-lab@Dasha-lab:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git add .
dasha-lab@Dasha-lab:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git commit -am 'feat(main): make course structure'
[master 3c8461f] feat(main): make course structure
 2 files changed, 1 insertion(+), 14 deletions(-)
 delete mode 100644 package.json
dasha-lab@Dasha-lab:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git push
Перечисление объектов: 5, готово.
Подсчет объектов: 100% (5/5), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (2/2), готово.
Запись объектов: 100% (3/3), 289 байтов | 289.00 КиБ/с, готово.
Всего 3 (изменений 1), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:dakozina/study_2024-2025_arh-pc.git
 d0fc60f..3c8461f master -> master

```

рис. 2.14: Добавление и сохранение изменений на сервере. Выгрузка изменений

Проверяем правильность выполнения работы на сайте GitHub (рис. 2.15).

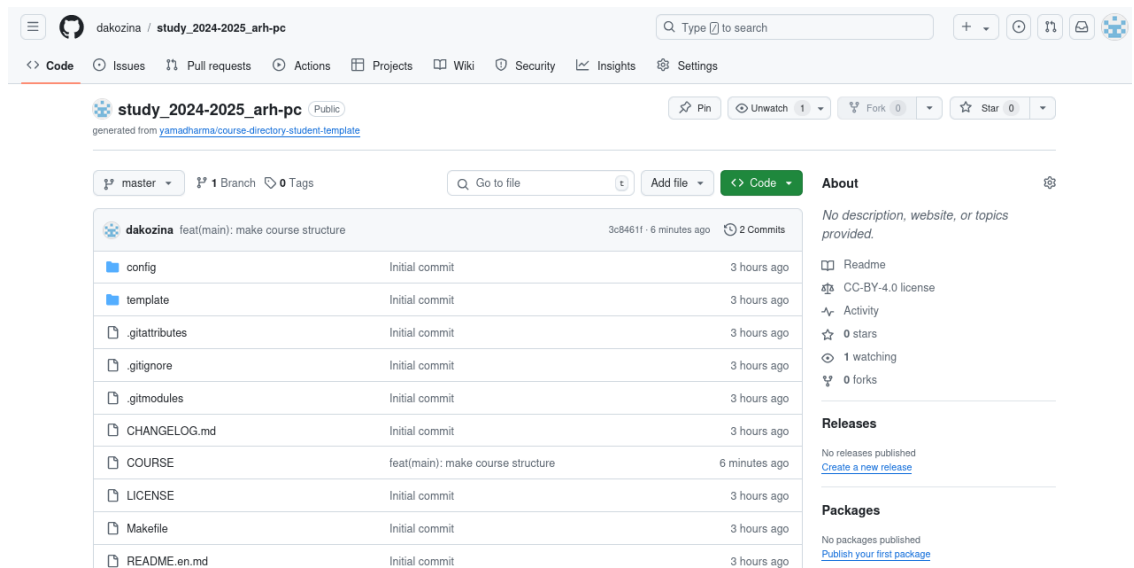


рис. 2.15: Страница репозитория

3 Выполнение заданий для самостоятельной работы

1. С помощью команды `cd` перейдем в директорию `labs/lab03/report`. С помощью команды `touch` создадим в каталоге файл для отчета по лабораторной работе № 2 (рис. 3.1).

```
dasha-lab@Dasha-lab:~$ cd ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc/labs/lab02/report
dasha-lab@Dasha-lab:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab02/report$ touch Л03_Козина_отчет
```

рис. 3.1: Создание файла

Оформить отчет мы сможем в текстовом редакторе LibreOffice Writer, найдя его в меню приложений. После открытия редактора открываем в нем созданный файл и начинаем работу над отчетом (рис. 3.2).

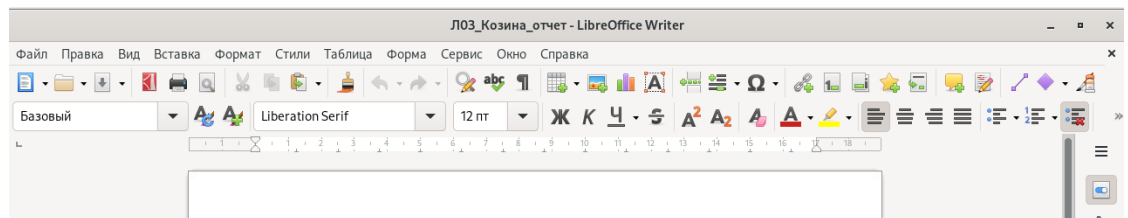


рис. 3.2: Работа над отчетом в текстовом редакторе

2. С помощью команды `cd` перейдем из подкаталога `lab03/report` в подкаталог `lab01/report` (рис. 3.3).

```
dasha-lab@Dasha-lab: ~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab02/report$ cd ..
dasha-lab@Dasha-lab: ~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab02$ cd ..
dasha-lab@Dasha-lab: ~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs$ cd lab01/report
dasha-lab@Dasha-lab: ~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab01/report$
```

рис. 3.3: Перемещение между директориями

С помощью команды `ср` копируем первую лабораторную из каталога «Загрузки» и проверяем правильность выполнения командой `ls` (рис. 3.4).

```
dasha-lab@Dasha-lab:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab01/report$ cp ~/Загрузки/Л01_Козина_отчет.pdf ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc/labs/lab01/report
dasha-lab@Dasha-lab:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab01/report$ ls
Л01_Козина_отчет.pdf
```

рис. 3.4: Копирование файла

3. С помощью команды `git add` добавляем в коммит созданный файл «Л01 Козина отчет» (рис. 3.5).

С помощью команды `git commit "..."` сохраняем изменения на сервере (рис. 3.6).

То же самое делаем для отчета по второй лабораторной работе.

С помощью команды `git push -f origin master` отправляем в центральный репозиторий сохраненные изменения (рис. 3.7).

Проверим на сайте GitHub правильность выполнения заданий (рис. 3.8).

4 Выводы