

# **Отчёт по лабораторной работе №2**

**Дисциплина: Архитектура компьютера**

Толстых Александра Андреевна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>7</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>10</b>
4.1	Настройка github . . . . .	10
4.2	Базовая настройка git . . . . .	11
4.3	Создание SSH-ключа . . . . .	12
4.4	Создание рабочего пространства и репозитория курса на основе шаблона . . . . .	13
4.5	Создание репозитория курса на основе шаблона . . . . .	14
4.6	Настройка каталога курса . . . . .	15
4.7	Задание для самостоятельной работы . . . . .	17
<b>5</b>	<b>Выводы</b>	<b>18</b>

## Список иллюстраций

4.1 Проблемы с регистрацией на сайте . . . . .	10
4.2 Созданная ранее учётная запись . . . . .	11
4.3 Выполнение команд для базовой настройки git . . . . .	12
4.4 Создание SSH-ключей . . . . .	12
4.5 Копирование открытого ключа из файла . . . . .	13
4.6 Добавление ключа через сайт . . . . .	13
4.7 Создание вложенных каталогов согласно структуре пространства	13
4.8 Использование шаблона . . . . .	14
4.9 Создание репозитория на основе шаблона . . . . .	14
4.10 Клонирование репозитория . . . . .	15
4.11 Удаление ненужных файлов . . . . .	15
4.12 Создание необходимых каталогов . . . . .	16
4.13 Комментирование и сохранение изменений . . . . .	16
4.14 Сравнение репозитория моего курса и шаблона . . . . .	16
4.15 Загрузка отчета первой лабораторной . . . . .	17
4.16 Создание файла для второй лабораторной . . . . .	17

## **Список таблиц**

# 1 Цель работы

Целью работы является изучить идеологию и применение средств контроля версий. Приобрести практические навыки по работе 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. Задание для самостоятельной работы

### 3 Теоретическое введение

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.

Приведем таблицу с наиболее часто используемыми командами `git` в таблице №1.

Команда	Описание
<code>git init</code>	Создание основного дерева репозитория
<code>git pull</code>	Получение обновлений (изменений) текущего дерева из центрального репозитория
<code>git push</code>	Отправка всех произведенных изменений локального дерева в центральный репозиторий
<code>git status</code>	Просмотр списка измененных файлов в текущей директории
<code>git diff</code>	Просмотр текущих изменений
<code>git add .</code>	Добавить все изменённые или созданные файлы или каталоги

Команда	
Команда	Описание
git add	Добавить конкретные изменённые или созданные файлы или каталоги
git rm	Удалить файл и/или каталог из индекса репозитория (при этом файл и каталог остаётся в локальной директории)
git commit -am 'Описание коммита'	Сохранить все добавленные изменения и все изменённые файлы
git checkout -b имя_ветки	Создание новой ветки, базирующейся на текущей
git checkout имя_ветки	Переключение на некоторую ветку (при переключении на ветку, которой ещё нет в локальном репозитории, она будет создана и связана с удалённой)
git push origin имя_ветки	Отправка изменений конкретной ветки в центральный репозиторий
git merge --no-ff имя_ветки	Слияние ветки с текущим деревом



---

Коман-	
да	Описание
<hr/>	
git branch -d имя_ветки	Удаление локальной уже слитой с основным деревом ветки
git branch -D имя_ветки	Принудительное удаление локальной ветки
git push origin :имя_ветки	Удаление ветки с центрального репозитория

---

## 4 Выполнение лабораторной работы

### 4.1 Настройка github

Сначала мы создаем учетную запись на сайте github и заполняем основные данные. Поскольку сайт не позволял регистрировать учетную запись с использованием корпоративной почты (рис. 4.1), я зашла на сайт используя свою уже существующую учетную запись на github, созданную ранее на личную почту (рис. 4.2).

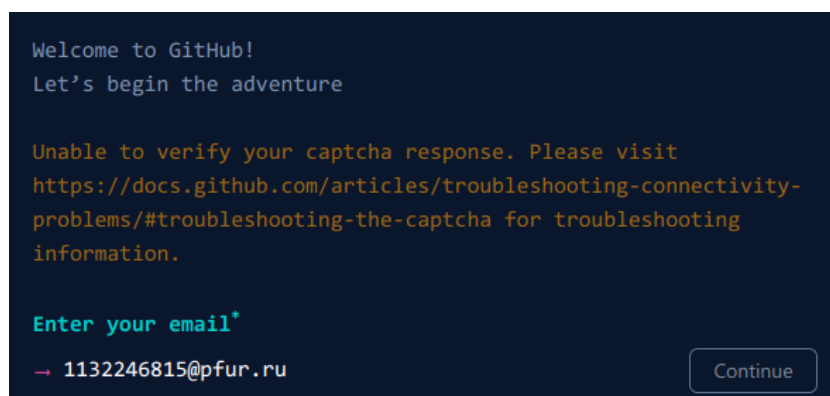


Рис. 4.1: Проблемы с регистрацией на сайте

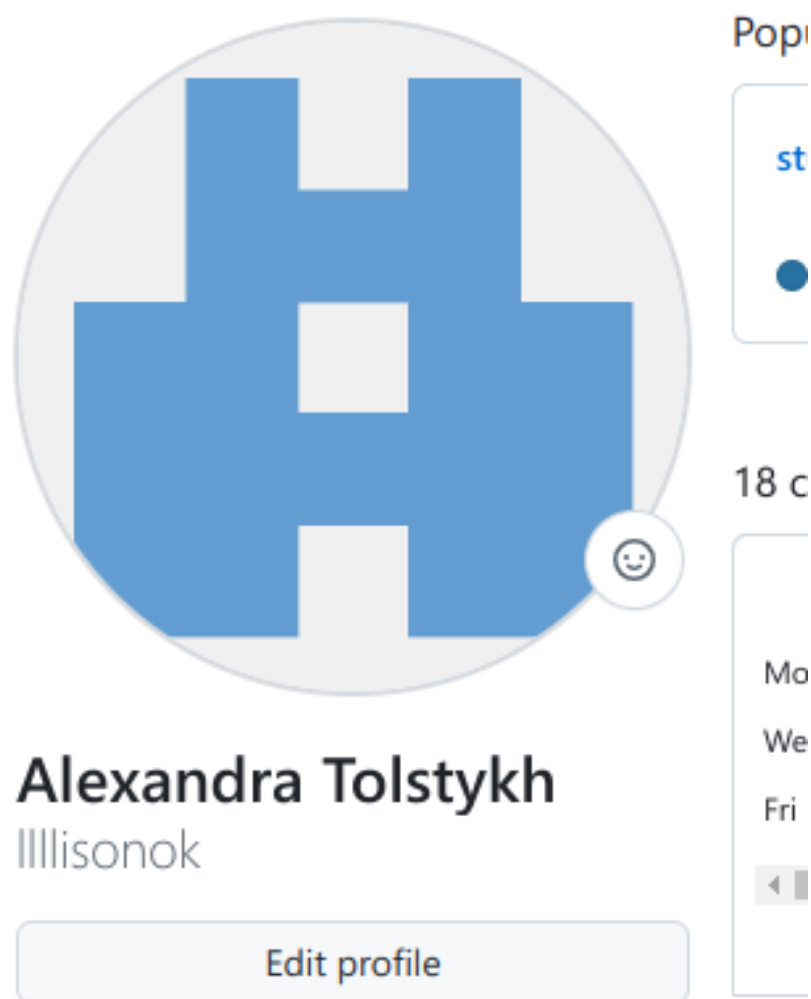


Рис. 4.2: Созданная ранее учётная запись

## 4.2 Базовая настройка git

Открываю терминал и делаю предварительную конфигурацию git (рис. 4.3). Ввожу команды «git config –global user.name» и «git config –global user.email», указывая данные от своего аккаунта. При помощи команды «git config –global core.quotePath false» настраиваю utf-8 в выводе сообщений git для их корректного отображения. Далее задаю имя начальной ветки «master», а также параметры «autocrlf»

(параметр необходим для настройки конвертации crlf в lf) и «safecrlf» (параметр необходим для проверки преобразования на обратимость).

```
aatolstihkh@dk5n51 ~ $ git config --global user.name "l1llisonok"
aatolstihkh@dk5n51 ~ $ git config --global user.email "shuratolstikh@mail.ru"
aatolstihkh@dk5n51 ~ $ git config --global core.quotePath false
aatolstihkh@dk5n51 ~ $ git config --global init.defaultBranch master
aatolstihkh@dk5n51 ~ $ git config --global core.autocrlf input
aatolstihkh@dk5n51 ~ $ git config --global core.safecrlf warn
aatolstihkh@dk5n51 ~ $
```

Рис. 4.3: Выполнение команд для базовой настройки git

## 4.3 Создание SSH-ключа

Для последующей идентификации пользователя на сервере репозитория необходимо сгенерировать приватный и открытый ключи. Ввожу команду «ssh-keygen -C» указывая как аргументы свои данные: имя, фамилию и почту (рис. 4.4). Ключи сохраняются в каталоге ~/.ssh/.

```
aatolstihkh@dk3n60 ~ $ ssh-keygen -C "Alexandra Tolstykh shuratolstikh@mail.ru"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/afs/.dk.sci.pfu.edu.ru/home/a/a/aatolstihkh/.ssh/id_ed25519): ~/.ssh/keys
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Passphrases do not match. Try again.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Saving key "~/.ssh/keys" failed: No such file or directory
aatolstihkh@dk3n60 ~ $ ssh-keygen -C "Alexandra Tolstykh shuratolstikh@mail.ru"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/afs/.dk.sci.pfu.edu.ru/home/a/a/aatolstihkh/.ssh/id_ed25519):
Created directory '/afs/.dk.sci.pfu.edu.ru/home/a/a/aatolstihkh/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /afs/.dk.sci.pfu.edu.ru/home/a/a/aatolstihkh/.ssh/id_ed25519
Your public key has been saved in /afs/.dk.sci.pfu.edu.ru/home/a/a/aatolstihkh/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:7EGnTPE2kPvE7b07KC0wpAMVVH0VtAbvOH+VSagAdxc Alexandra Tolstykh shuratolstikh@mail.ru
The key's randomart image is:
+--[ED25519 256]--+
|      .o+=oo+oE.  |
|      . +*.oo..   |
|      . o+=+. .   |
|      . =o++=o . o|
|      . oSooo.. o.|
|      o.o..o. . . |
|      ..o . . . . |
|      o o.o .     |
|      o .o .     |
+-----[SHA256]-----+
```

Рис. 4.4: Создание SSH-ключей

Копируем ключ в буфер обмена при помощи команды «cat ~/.ssh/id\_rsa.pub | xclip -sel clip» (рис. 4.5).

```
aatolstihkh@dk3n60 ~ $ ls ~/.ssh/  
id_ed25519 id_ed25519.pub  
aatolstihkh@dk3n60 ~ $ cat ~/.ssh/id_ed25519.pub | xclip -sel clip  
aatolstihkh@dk3n60 ~ $
```

Рис. 4.5: Копирование открытого ключа из файла

Вставляю скопированный ключ в поле на сайте, чтобы добавить его (рис. 4.6).

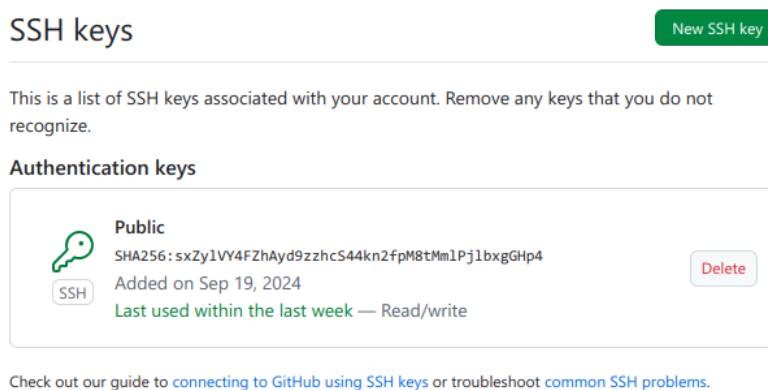


Рис. 4.6: Добавление ключа через сайт

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

При выполнении лабораторных работ следует придерживаться структуры рабочего пространства, поэтому при помощи команды `mkdir` с аргументом `-p` создаю нужную вложенную последовательность каталогов (рис. 4.7).

```
aatolstihkh@dk3n60 ~ $ mkdir -p ~/work/study/2024-2025/"Архитектура компьютера"  
aatolstihkh@dk3n60 ~ $
```

Рис. 4.7: Создание вложенных каталогов согласно структуре пространства

## 4.5 Создание репозитория курса на основе шаблона

Репозиторий на основе шаблона можно создать через web-интерфейс github. Для этого перехожу на страницу репозитория с шаблоном курса по ссылке, указанной в лабораторной работе. Далее использую кнопку «Use this template» (рис. 4.8).

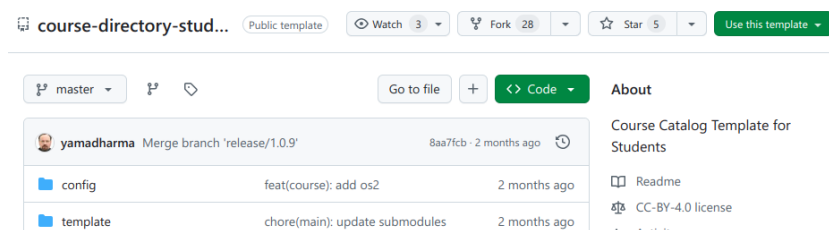


Рис. 4.8: Использование шаблона

В открывшемся окне задаю имя репозитория «study\_2024-2025\_arhpc» и создаю его при помощи кнопки «Create repository from template» (рис. 4.9).

**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**

yamadharm/course-directory-student-template

Start your repository with a template repository's contents.

☐ **Include all branches**  
Copy all branches from yamadharm/course-directory-student-template and not just the default branch.

**Owner \*** illisonok / **Repository name \*** study\_2024-2025\_arhpc

✓ Your new repository will be created as study\_2024-2025\_arhpc.  
The repository name can only contain ASCII letters, digits, and the characters ., -, and \_.

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

**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. 4.9: Создание репозитория на основе шаблона

Через терминал перехожу в созданный каталог курса при помощи команды `cd`. Клонировать данный репозиторий при помощи команды «`git clone –recursive git@github.com:/lllisonok/study_2024-2025_arhpc.git`» (рис. 4.10).

```
aatolstikh@dk2n21 ~ $ cd ~/work/study/2024-2025/"Архитектура компьютера"
aatolstikh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера $ git clone --recursive git@github.com:
lllisonok/study_2024-2025_arhpc.git
Клонирование в «study_2024-2025_arhpc»...
remote: Enumerating objects: 33, done.
remote: Counting objects: 100% (33/33), done.
remote: Compressing objects: 100% (32/32), done.
remote: Total 33 (delta 1), reused 18 (delta 0), pack-reused 0 (from 0)
Получение объектов: 100% (33/33), 18.83 КиБ | 448.00 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) запе
гистрирован по пути «template/report»
Клонирование в «/afs/.dk.sci.pfu.edu.ru/home/a/a/aatolstikh/work/study/2024-2025/Архитектура компьютера
/study_2024-2025_arhpc/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)
Получение объектов: 100% (111/111), 102.17 КиБ | 147.00 КиБ/с, готово.
Определение изменений: 100% (42/42), готово.
Клонирование в «/afs/.dk.sci.pfu.edu.ru/home/a/a/aatolstikh/work/study/2024-2025/Архитектура компьютера
/study_2024-2025_arhpc/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)
Получение объектов: 100% (142/142), 341.09 КиБ | 2.34 МиБ/с, готово.
Определение изменений: 100% (60/60), готово.
Submodule path 'template/presentation': checked out 'c9b2712b4b2d431ad5086c9c72a02bd2fca1d4a6'
Submodule path 'template/report': checked out 'c26e22effe7b3e0495707d82ef561ab185f5c748'
aatolstikh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера $
```

Рис. 4.10: Клонирование репозитория

## 4.6 Настройка каталога курса

Перехожу в каталог курса при помощи команды `cd`, а затем при помощи коман-  
ды `rm` удаляю ненужные файлы (рис. 4.11).

```
aatolstikh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера $ cd ~/work/study/2024-2025/"Архитектур
а компьютера"/study_2024-2025_arhpc
aatolstikh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc $ rm package.json
```

Рис. 4.11: Удаление ненужных файлов

Далее создаю необходимые каталоги так, как это указано в задании (рис. 4.12).

```

aato1stihkh@dk3n60 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc $ echo arch-pc >
COURSE
aato1stihkh@dk3n60 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc $ make
Usage:
  make <target>

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

aato1stihkh@dk3n60 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc $ make prepare

```

Рис. 4.12: Создание необходимых каталогов

Используя команды «git add .», «git commit -am», «git push» комментирую и сохраняю внесенные изменения (рис. 4.13).

```

aato1stihkh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc $ git add .
aato1stihkh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc $ git commit -am
'feat(main): make course structure'
[master 44fd5ef] feat(main): make course structure
2 files changed, 1 insertion(+), 14 deletions(-)
delete mode 100644 package.json
aato1stihkh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc $ git push
Перечисление объектов: 5, готово.
Подсчет объектов: 100% (5/5), готово.
При сжатии изменений используется до 6 потоков
Сжатие объектов: 100% (2/2), готово.
Запись объектов: 100% (3/3), 280 байтов | 280.00 КиБ/с, готово.
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:llllisonok/study_2024-2025_arhpc.git
 e197c9e..44fd5ef master -> master
aato1stihkh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc $

```

Рис. 4.13: Комментирование и сохранение изменений

Затем открываю сайт и проверяю, что все выполнилось корректно (рис. 4.14). Для этого сравниваю папки своего репозитория с шаблоном, отличие должно быть в двух файлах: «package.json» и «labs». Это действительно так. Значит каталог курса настроен корректно.

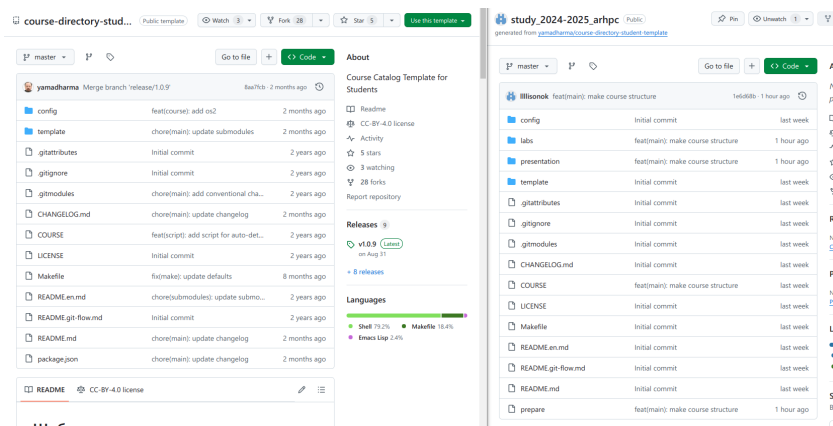


Рис. 4.14: Сравнение репозитория моего курса и шаблона



## 4.7 Задание для самостоятельной работы

Сначала загружу отчет о выполнении первой лабораторной (рис. 4.15). Для этого сохраню его в папку «Документы», а затем скопирую в папку «lab01». Затем комментирую и сохраняю изменения.

```
aatolstikh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc/labs/lab01 $ cp ~/Документы/Л01_Толстых_отчет.pdf ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc/labs/lab01
aatolstikh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc/labs/lab01 $ ls
Л01_Толстых_отчет.pdf
aatolstikh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc/labs/lab01 $ git add Л01_Толстых_отчет.pdf
aatolstikh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc/labs/lab01 $ git commit -am "Add lab's file"
[master cdf2e9b] Add lab's file
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 labs/lab01/Л01_Толстых_отчет.pdf
aatolstikh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc/labs/lab01 $ git push
```

Рис. 4.15: Загрузка отчета первой лабораторной

Далее, чтобы убедиться в приобретенных навыках, перехожу в папку для второй лабораторной и при помощи команды `touch` создаю файл для отчета данной (второй) лабораторной работы (рис. 4.16). Комментирую и сохраняю изменения. (Данный файл позже будет заменен, поскольку отчеты о лабораторных я пишу дома, действия выполняются лишь для того чтобы убедиться, что необходимые команды изучены).

```
aatolstikh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc/labs $ cd lab02
aatolstikh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc/labs/lab02 $ touch Л02_Толстых_отчет.pdf
aatolstikh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc/labs/lab02 $ git add Л02_Толстых_отчет.pdf
aatolstikh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc/labs/lab02 $ git commit -am "Add lab's file"
[master 450d904] Add lab's file
1 file changed, 0 insertions(+), 0 deletions(-)
rename labs/( => lab02)/Л02_Толстых_отчет.pdf (100%)
aatolstikh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc/labs/lab02 $ git push
Перечисление объектов: 100% (6/6), готово.
При скатии изменений используется до 6 потоков
Сжатие объектов: 100% (4/4), готово.
Запись объектов: 100% (4/4), 354 байта | 354.00 КиБ/с, готово.
Total 4 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
aatolstikh@dk2n21 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc/labs/lab02 $
```

Рис. 4.16: Создание файла для второй лабораторной

Затем открываю сайт и проверяю наличие отчета о первой лабораторной работе, а также файла для отчета о второй лабораторной работе. Все выполнено корректно. Далее завершаю написание отчета и через сайт заменяю файл с отчетом второй лабораторной.

## 5 Выводы

В результате выполнения данной лабораторной работы я изучила идеологию и применение средств контроля версий, а также приобрела практические навыки по работе с системой git.