

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: Грибашова Анастасия

Группа: НКАбд-02-24

МОСКВА
2024 г.

Оглавление:

1. Цель работы.....	3
2. Задание.....	4
3. Теоретическое введение.....	5
4. Выполнение лабораторной работы.....	7
5. Задание для самостоятельной работы.....	14
6. Выводы.....	16
Список литературы.....	17

1. Цель работы.

Изучение идеологии, применение средств контроля версий и обретение практических навыков по работе с системой git.

2. Задание.

Используя методические материалы, провести работу, состоящую из настройки git и создания репозитория.

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 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.

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.

Таблица 3.1. Основные команды git

Команда	Описание
git init	создание основного дерева репозитория
git pull	получение обновлений (изменений) текущего дерева из центрального репозитория
git push	отправка всех произведённых изменений локального дерева в центральный репозиторий
git status	просмотр списка изменённых файлов в текущей директории
git diff	просмотр текущих изменения
git add .	добавить все изменённые и/или созданные файлы и/или каталоги
git add имена_файлов	добавить конкретные изменённые и/или созданные файлы и/или каталоги
git rm имена_файлов	удалить файл и/или каталог из индекса репозитория
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
git push origin имя_ветки	отправка изменений конкретной ветки в центральный репозиторий
git merge --no-ff имя_ветки	слияние ветки с текущим деревом
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. Техническое обеспечение

Лабораторная работа была выполнена на домашнем компьютере под управлением ОС Fedora Workstation 40.

4.2. Настройка github

Чтобы начать выполнение работы, создаю аккаунт на GitHub и заполняю основные данные (рис. 4.2.1)

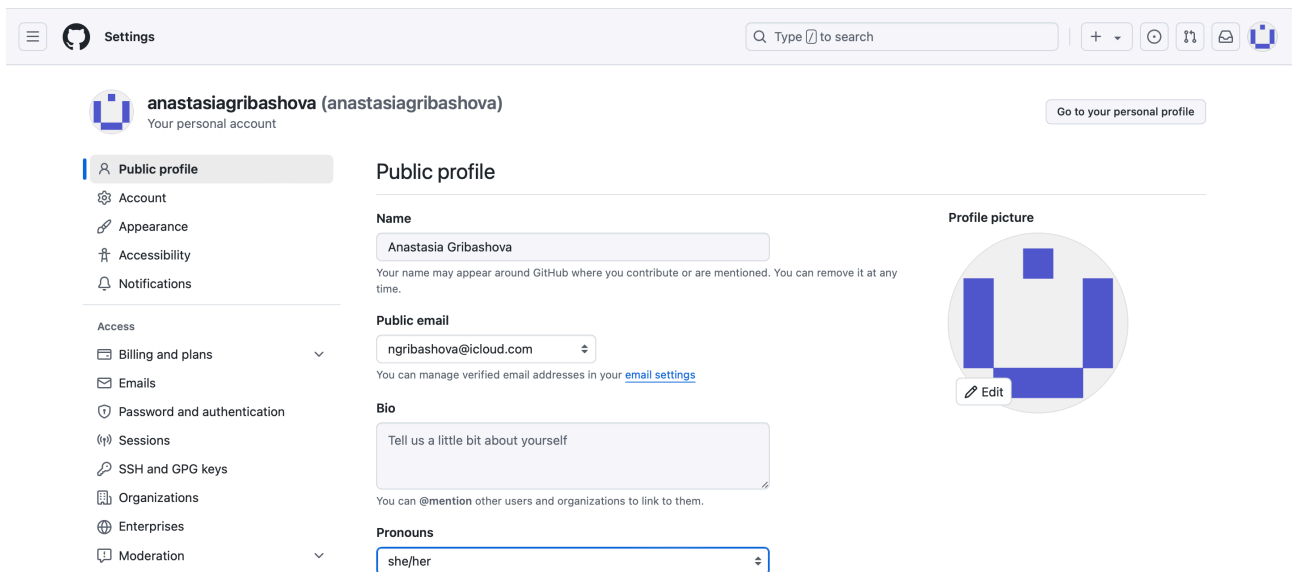


Рис. 4.2.1. Создание аккаунта на GitHub.

4.3. Базовая настройка git

Открываю терминал и делаю предварительную конфигурацию git (рис. 4.3.1)

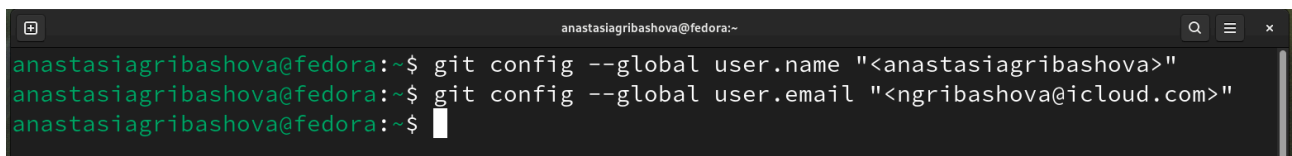


Рис. 4.3.1. Создание предварительной конфигурации git

Затем настраиваю utf-8 в выводе сообщений git (рис. 4.3.2)



Рис. 4.3.2. Настройка utf-8 в выводе сообщений git

Далее задаю имя начальной ветки, после чего ввожу параметры autocrlf и safecrlf (рис. 4.3.3)

```
anastasiagribashova@fedora:~$ git config --global user.name "<anastasiagribashova>"
anastasiagribashova@fedora:~$ git config --global user.email "<ngribashova@icloud.com>"
anastasiagribashova@fedora:~$ git config --global core.quotepath false
anastasiagribashova@fedora:~$ git config --global init.defaultBranch master
anastasiagribashova@fedora:~$ git config --global core.autocrlf input
anastasiagribashova@fedora:~$ git config --global core.safecrlf warn
anastasiagribashova@fedora:~$
```

Рис. 4.3.3. Задача имени начальной ветки, а также параметров autocrlf и safecrlf

4.4. Создание SSH ключа

Генерирую пару ключей(приватный и открытый) для последующей идентификации пользователя на сервере репозитория и копирую из локальной консоли в буфер обмена (рис. 4.4.1)

```
anastasiagribashova@fedora:~$ ssh-keygen -C "anastasiagribashova <ngribashova@icloud.com>"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/anastasiagribashova/.ssh/id_ed25519):
/home/anastasiagribashova/.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 /home/anastasiagribashova/.ssh/id_ed25519
Your public key has been saved in /home/anastasiagribashova/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:6Rgk2og2GvDnVn5Qt9gY1xU/9yxwgJpDgo4dQIRWgWo anastasiagribashova <ngribashova@icloud.com>
The key's randomart image is:
+--[ED25519 256]--+
|  ++..      .. o. |
| .o o . . . o . |
|+  +...oooo o ..o|
|oE.+oo .+O . o .+|
|+o+ o + S.o . o |
|o..o o =         .|
|.  o o o         |
| .  .           |
|                  |
+----[SHA256]-----+
anastasiagribashova@fedora:~$ cat ~/.ssh/id_ed25519 | xclip -sel clip
bash: cat ~/.ssh/id_ed25519: Нет такого файла или каталога
anastasiagribashova@fedora:~$ cat ~/.ssh/id_ed25519.pub | xclip -sel clip
anastasiagribashova@fedora:~$
```

Рис. 4.4.1. Создание пары ключей и копирование её в буфер обмена

Затем захожу на GitHub и вставляю SSH ключ в нужное поле на сайте (рис. 4.4.2)

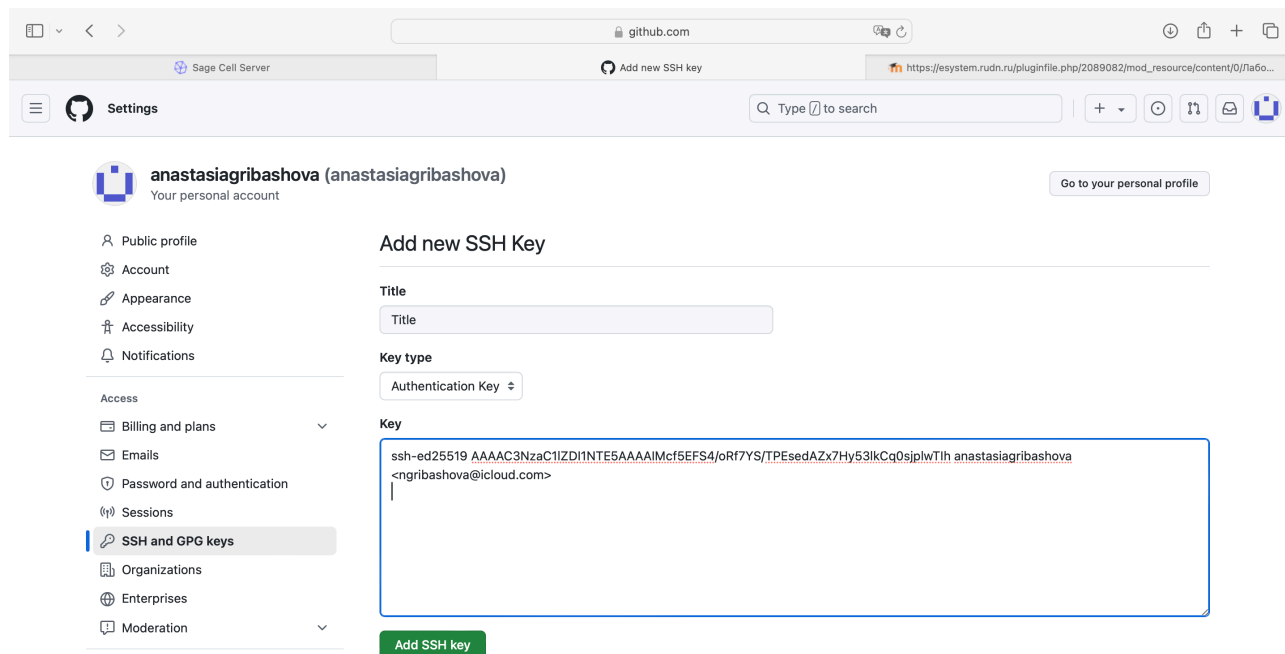


Рис. 4.4.2. Ввод SSH ключа на сайт GitHub

Проверяю правильность работы ключа на сайте (рис. 4.4.3)

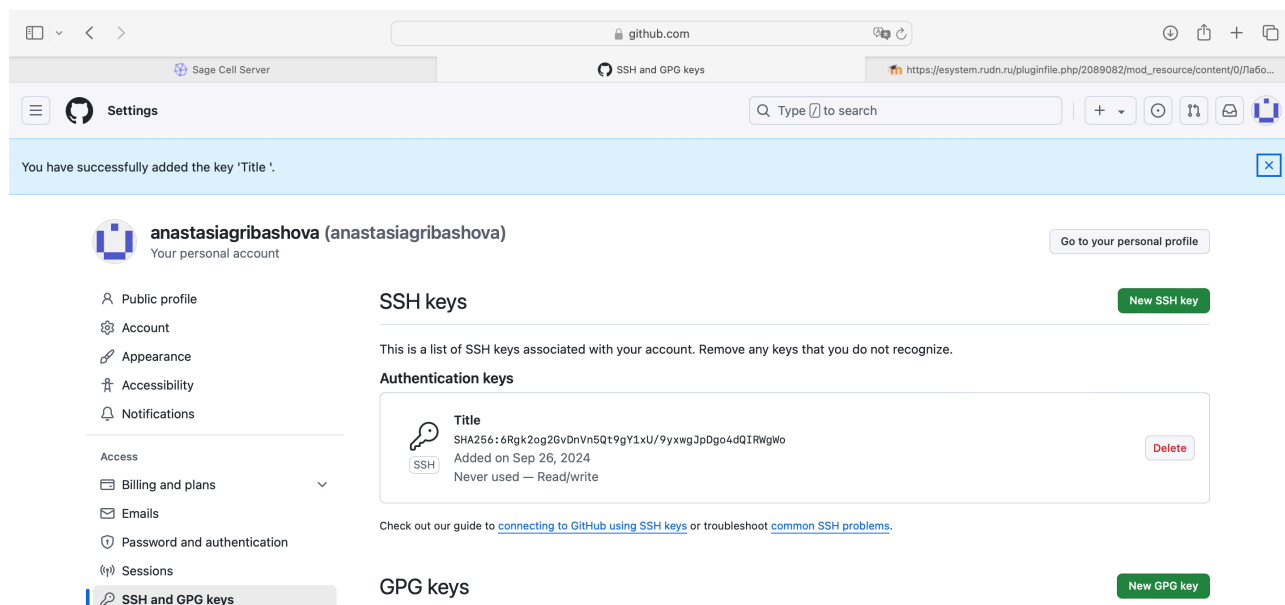


Рис. 4.4.3. Проверка правильности работы ключа

4.5. Создание рабочего пространства и репозитория курса на основе шаблона

Открываю терминал и создаю каталог для предмета «Архитектура компьютера» (рис. 4.5.1)

```
anastasiagribashova@fedora:~$ mkdir -p ~/work/study/2024-2025/"Архитектура компьютера"
anastasiagribashova@fedora:~$
```

Рис. 4.5.1. Создание каталога для предмета «Архитектура компьютера»

4.6. Создание репозитория курса на основе шаблона

Перехожу на страницу репозитория с шаблоном курса по ссылке, указанном в инструкции к лабораторной работе, нажимаю кнопку «use this temple», задаю имя репозитория и создаю его (рис. 4.6.1)

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 * **Repository name ***

anastasiagribashova / study_2024-2025_arh-pc

✓ **Your new repository will be created as study_2024-2025_arh-pc.**
The repository name can only contain ASCII letters, digits, and the characters -, ., and _.

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

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. 4.6.1. Создание репозитория

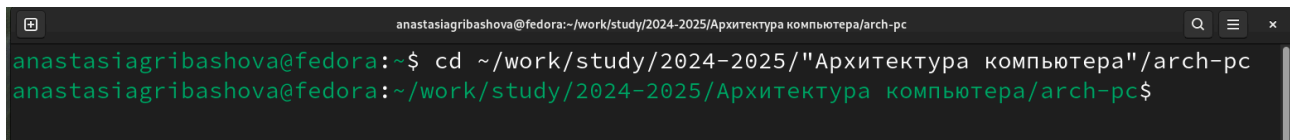
Далее открываю терминал, перехожу в каталог курса и копирую созданный репозиторий с помощью команды git clone (рис. 4.6.2)

```
anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера
anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера$ git clone --recursive git@github.com:anastasiagribashova/study_2024-2025_arh-pc.git arch-pc
Клонирование в «arch-pc»...
remote: Enumerating objects: 34, done.
remote: Counting objects: 100% (34/34), done.
remote: Compressing objects: 100% (33/33), done.
remote: Total 34 (delta 1), reused 18 (delta 0), pack-reused 0 (from 0)
Получение объектов: 100% (34/34), 19.58 КиБ | 274.00 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/anastasiagribashova/work/study/2024-2025/Архитектура компьютера/arch-pc/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 | 462.00 КиБ/с, готово.
Определение изменений: 100% (42/42), готово.
Клонирование в «/home/anastasiagribashova/work/study/2024-2025/Архитектура компьютера/arch-pc/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 | 1003.00 КиБ/с, готово.
Определение изменений: 100% (60/60), готово.
Submodule path 'template/presentation': checked out 'c9b2712b4b2d431ad5086c9c72a02bd2fca1d4a6'
Submodule path 'template/report': checked out 'c26e22effe7b3e0495707d82ef561ab185f5c748'
anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера$
```

Рис. 4.6.2. Клонирование репозитория

4.7. Настройка каталога курса

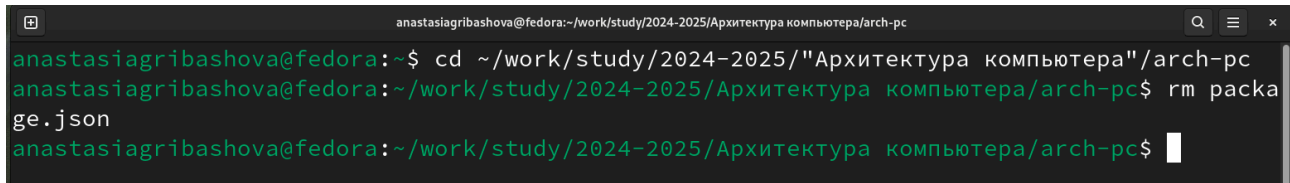
Перехожу в каталог курса (рис. 4.7.1)



```
anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc
anastasiagribashova@fedora:~$ cd ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc
anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$
```

Рис. 4.7.1. Каталог курса

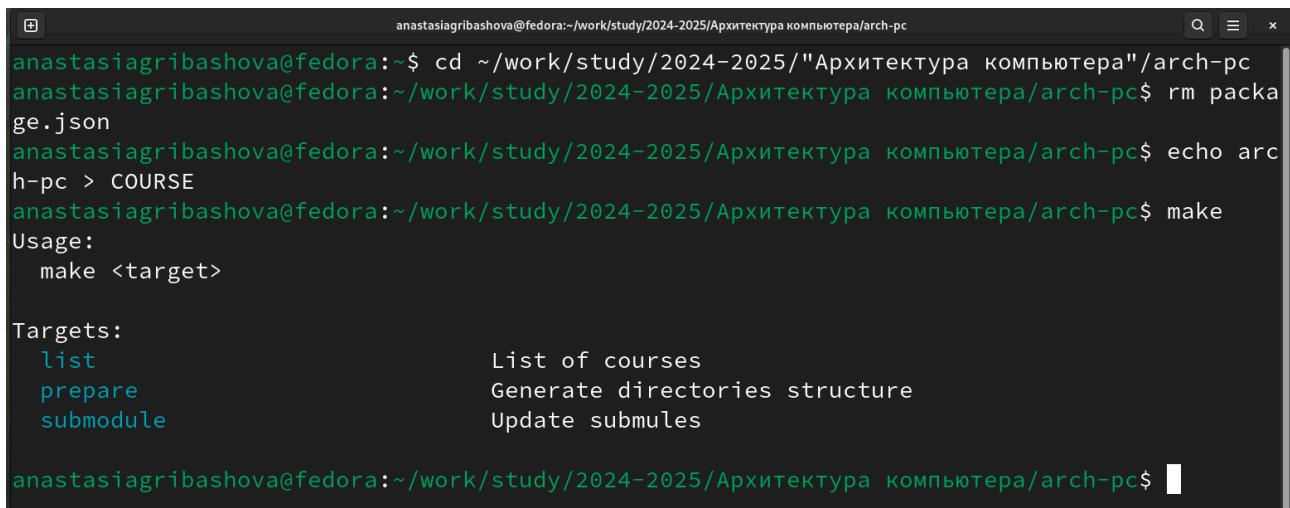
Удаляю лишние файлы с помощью команды `rm` (рис. 4.7.2)



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

Рис. 4.7.2

Создаю необходимые каталоги (рис. 4.7.3)



```
anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc
anastasiagribashova@fedora:~$ cd ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc
anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ rm package.json
anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ echo arch-pc > COURSE
anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ make
Usage:
  make <target>

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

anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$
```

Рис. 4.7.3

Отправляю файлы на сервер (рис. 4.7.4)

```
anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$
anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ rm package.json
anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ echo arch-pc > COURSE
anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ make
Usage:
  make <target>

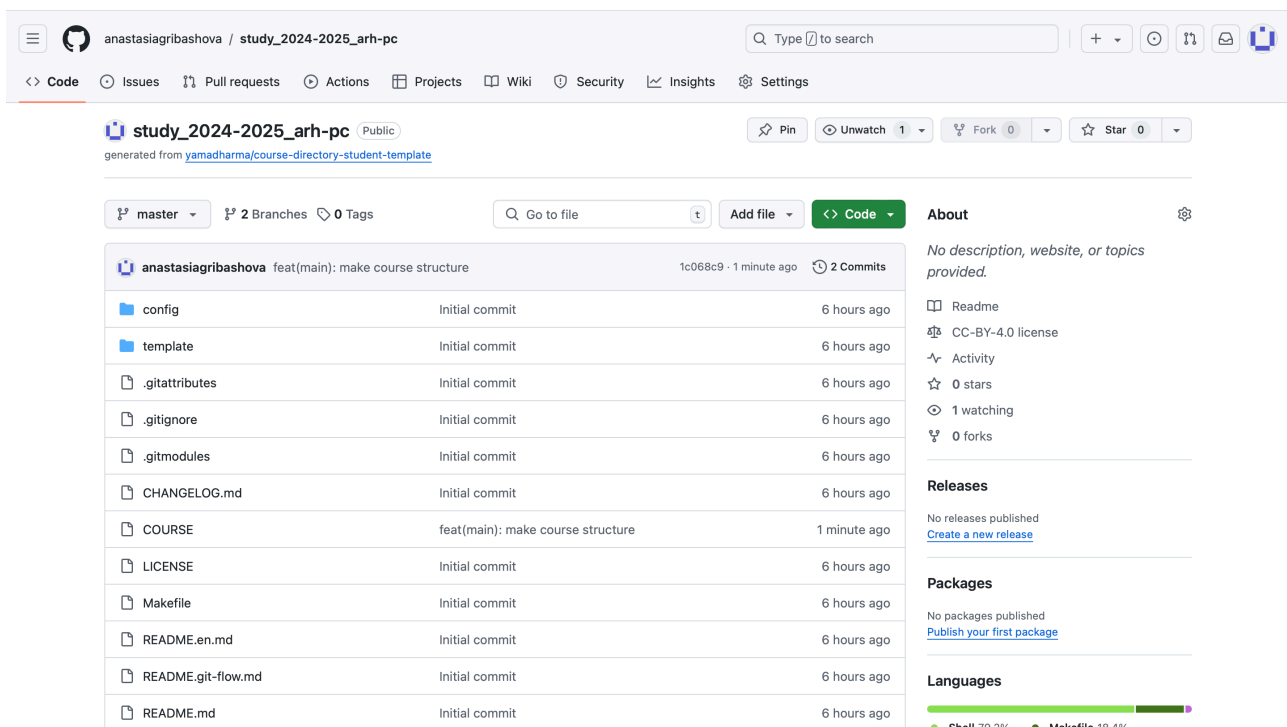
Targets:
  list           List of courses
  prepare       Generate directories structure
  submodule     Update submules

anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git add.
git: «add.» не является командой git. Смотрите «git --help».

Самые похожие команды:
  add

anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git add
.
anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$ git commit -am 'feat(main): make course structure'
[master 1c068c9] feat(main): make course structure
 2 files changed, 1 insertion(+), 14 deletions(-)
 delete mode 100644 package.json
anastasiagribashova@fedora:~/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), 287 байтов | 287.00 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:anastasiagribashova/study_2024-2025_arh-pc.git
   03f3fa2..1c068c9  master -> master
anastasiagribashova@fedora:~/work/study/2024-2025/Архитектура компьютера/arch-pc$
```

Рис. 4.7.4. Отправление файлов на сервер
В конце проверяю правильность на странице github (рис. 4.7.5)



5. Задания для самостоятельной работы.

Задание 1.

Создаю каталоги `~/work/study/2024-2025/«Архитектура компьютера»/arch-pc/labs/lab01/report` и `~/work/study/2024-2025/«Архитектура компьютера»/arch-pc/labs/lab02/report` с помощью команды `mkdir` (рис. 5.1.1), после чего создаю отчет по выполнению лабораторной работы в каталоге `labs/lab02/report` (рис. 5.1.2)

A terminal window titled 'anastasiagribashova@fedora:~' with search, menu, and close icons in the top right. It shows three lines of terminal output: the first two lines show the successful execution of 'mkdir -p' commands to create the directory structure for lab01 and lab02 reports, and the third line shows the prompt after the second command.

```
anastasiagribashova@fedora:~$ mkdir -p ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc/labs/lab01/report
anastasiagribashova@fedora:~$ mkdir -p ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc/labs/lab02/report
anastasiagribashova@fedora:~$
```

Рис. 5.1.1.

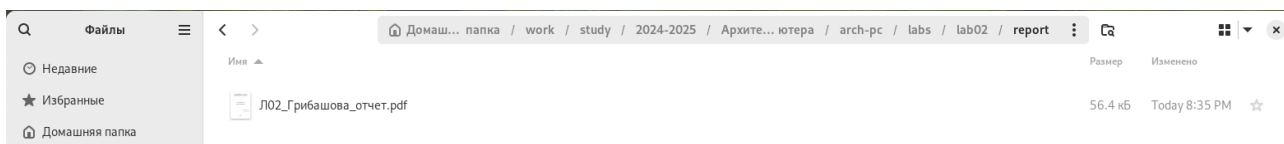


Рис. 5.1.2.

Задание 2.

Копирую отчеты по выполнению предыдущей лабораторной работы в каталог `labs/lab01/report` (рис. 5.2)

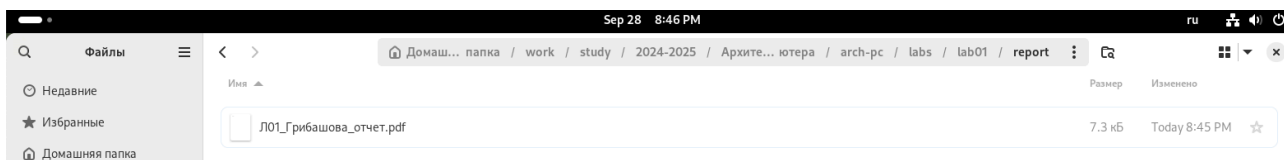


Рис. 5.2.

Задание 3.

Загружаю файлы на github (рис. 5.3)

6. Выводы

При выполнении данной лабораторной работы я приобрела навыки по работы с системой git, а также изучила идеологию и применение средств контроля версий.

Литература

1. <https://esystem.rudn.ru/mod/resource/view.php?id=1030550>
2. <https://esystem.rudn.ru/mod/page/view.php?id=1030492>
3. <https://esystem.rudn.ru/mod/resource/view.php?id=1030496>
4. github.com