

# **Лабораторная работа №2**

**Простейший вариант**

Тойчубекова Асель Нурлановна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Выводы</b>	<b>18</b>

## Список иллюстраций

2.1	Аккаунт на Github . . . . .	7
2.2	Конфигурация git. Настройка utf-8. Формировка названия начальной ветки. Формировка параметра . . . . .	8
2.3	Генерирование ключей. . . . .	8
2.4	Сгенерированный ключ в github . . . . .	9
2.5	Структура рабочего пространства . . . . .	9
2.6	Репозиторий, созданный по шаблону . . . . .	10
2.7	Репозиторий, созданный по шаблону . . . . .	10
2.8	Удаление ненужного файла. Создание каталога . . . . .	11
2.9	Добавление указанных файлов в индекс. . . . .	11
2.10	Фиксирование добавленных в индекс изменений. . . . .	12
2.11	Фиксирование добавленных в индекс изменений. . . . .	13
2.12	Проверка иерархии в рабочем пространстве . . . . .	14
2.13	Проверка иерархии в рабочем пространстве . . . . .	15
2.14	Рис.6.1 Загрузка лабораторной работы №1 в репозиторий. . . . .	16
2.15	Рис.6.2 Проверка загрузки в Github . . . . .	16
2.16	Рис.6.3 Загрузка в github лабораторной работы№2 . . . . .	16
2.17	Рис.6.4 Проверка загрузки лабораторной работы№2 в github . . . . .	17

## Список таблиц

# 1 Цель работы

Целью работы является изучить идеологию и применение средств контроля версий. Приобрести практические навыки по работе с системой git. Изучить общие понятия систем контроля версии, систему GIT, научиться использовать основные команды git, работать с системой при наличии центрального репозитория, базовым настройкам github и git, создавать SSH-ключи, создавать рабочее пространство.

## 2 Задание

Ознакомиться с системой контроля версии GIT. Изучить: -основные команды git, -работу с системой при наличии центрального репозитория, -настройки github, -базовые настройки git, -создание SSH-ключи, -Создание рабочего пространства и репозитория курса на основе шаблона, -Создание репозитория курса на основе шаблона, который был предоставлен в ходе лабораторной работы №2, -как настроить каталог курса. Задание для самостоятельной работы: -Создать отчет по выполнению лабораторной работы в соответствующем каталоге рабочего пространства (labs>lab01>report). 2.Скопировать отчеты по выполнению предыдущих лабораторных работ, то есть лабораторную работу№1, в соответствующие каталоги созданного рабочего пространства. 3.Загрузить файлы на github. # Теоретическое введение Системы контроля версий (Version Control Systems, VCS) — это инструменты, которые позволяют отслеживать изменения в исходном коде и других файлах проекта, а также облегчают коллаборацию между разработчиками. Они предоставляют возможность возвращаться к предыдущим версиям кода, сравнивать и сливать изменения, а также разрешать конфликты в коде. В классических системах контроля версий используется централизованная модель, предполагающая наличие единого репозитория для хранения файлов. Функции системы контроля версий: • Хранение несколько версий одного и того же документа (история версий), • Хранение истории разработки, • При необходимости возвращение к более ранним версиям документа (отмена изменений); • определение, кто и когда сделал изменение (поиск «виновного»), • совмещение изменений сделанных разными разработчиками (синхронизация работы коман-

ды) • реализация альтернативных/экспериментальных вариантов проекта. Среди классических VCS наиболее известны CVS, Subversion, а среди распределённых — Git, Bazaar, Mercurial. Принцип их работы схож, они отличаются лишь синтаксисом используемых в работе команд. Контроль версии Git — представляет собой набор программ командной строки, это распределенная система контроля версий, что означает, что каждый разработчик имеет свою собственную копию репозитория с полной историей изменений. Рассмотрим основные команды для работы с Git (рис. ??).

[Основные команды для работы с Git](image/РИС111(2.png){#fig:001 width=70% }

При наличии центрального репозитория пользователь может начать работу с проверки и получения изменений из центрального репозитория, а затем уже вносить изменения в локальное дерево или ветки, а дальше после всех изменений можно сохранить и отправить изменения в центральный репозиторий. # Выполнение лабораторной работы **Настройка GITHUB**. Для начала работы нам нужно настроить github. Создадим учётную запись на сайте <https://github.com/> и заполним основные данные. После всего заполнения у нас появится свой аккаунт, где мы можем создавать свои репозитории и работать над своими проектами. Созданный аккаунт показан на (рис. 2.1)

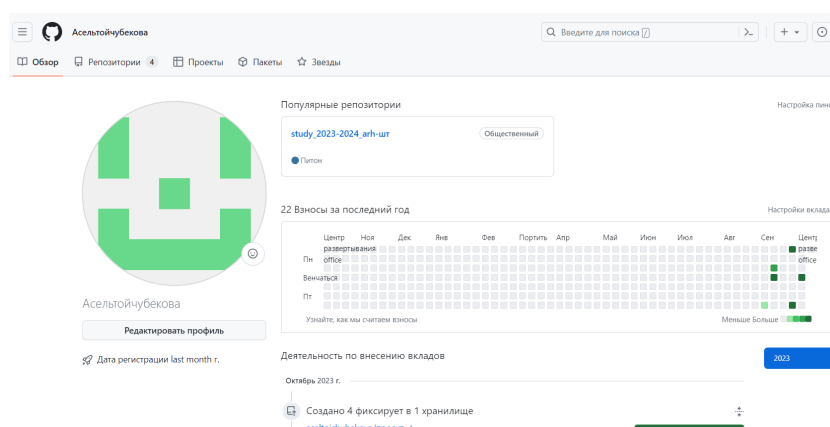


Рис. 2.1: Аккаунт на Github

**Базовая настройка Git.** Сперва сделаем предварительную конфигурацию git. Открыв терминал используя команды git такие как: `git config --global`, указываем свое имя, как в аккаунте в github, и свой email. Далее требуется подождать некоторое время, а после выполнении конфигурации настроим utf-8 в выводе сообщений git. Далее зададим имя начальной ветки, будем называть ее “master”. Также зададим параметры. Результат проделанной работы можно наблюдать на (рис. 2.2)

```
antoyjchubekova@dk6n55 ~ $ git config --global user.name "<aseltoichubekova>"
antoyjchubekova@dk6n55 ~ $ git config --global user.email "<aseltoychubekova714@gmail.com>"
antoyjchubekova@dk6n55 ~ $ git config --global core.quotepath false
antoyjchubekova@dk6n55 ~ $ git config --global init.defaultBranch master
antoyjchubekova@dk6n55 ~ $ git config --global core.autocrlf input
antoyjchubekova@dk6n55 ~ $ git config --global core.safecrlf warm
```

Рис. 2.2: Конфигурация git. Настройка utf-8. Формировка названия начальной ветки. Формировка параметра .

**Создание SSH-ключа.** Для последующей идентификации пользователя на сервере репозитория сгенерируем пару ключей (приватный и открытый). Далее зайдя на сайт github под своей учетной записью перейдем в настройки и выберем в боковом меню SSH and GPG keys, нажав кнопку New SSH key и вставив на указанном поле скопированный из консоли, с помощью команды `cat`, ключ загрузим сгенерированный открытый ключ. Укажем имя Title для ключа. Проведенную работу в командной строке мы видим на (рис. 2.3)

```
antoyjchubekova@dk6n55 ~ $ ssh-keygen -C "asel toichubekova <aseltoychubekova714@gmail.com>"
Generating public/private rsa key pair.
Enter file in which to save the key (/afs/.dk.sci.pfu.edu.ru/home/a/n/antoyjchubekova/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /afs/.dk.sci.pfu.edu.ru/home/a/n/antoyjchubekova/.ssh/id_rsa
Your public key has been saved in /afs/.dk.sci.pfu.edu.ru/home/a/n/antoyjchubekova/.ssh/id_rsa.pub
The key's fingerprint is:
SHA256:wpXzaeS74dBUfh8JqupQztKBB1PGYn/PIFkJrq5a4Ak asel toichubekova <aseltoychubekova714@gmail.com>
The key's randomart image is:
+----[RSA 3072]-----+
|  .+.  .  |
| o+. o .  |
| .oo.o +. . |
| +=. + =+. . |
|E . o=S *. . o |
|o.o . =.*. . . |
| o. o +..+ .  |
| .. o .o o  |
|o. .o o  |
+----[SHA256]-----+
antoyjchubekova@dk6n55 ~ $ cat ~/.ssh/id_rsa.pub | xclip -sel clip
```

Рис. 2.3: Генерирование ключей.



Можно убедиться, что все сделано правильно, зайдя в Github, перезагрузить аккаунт, и увидеть, что ключ загружен. (рис. 2.4)

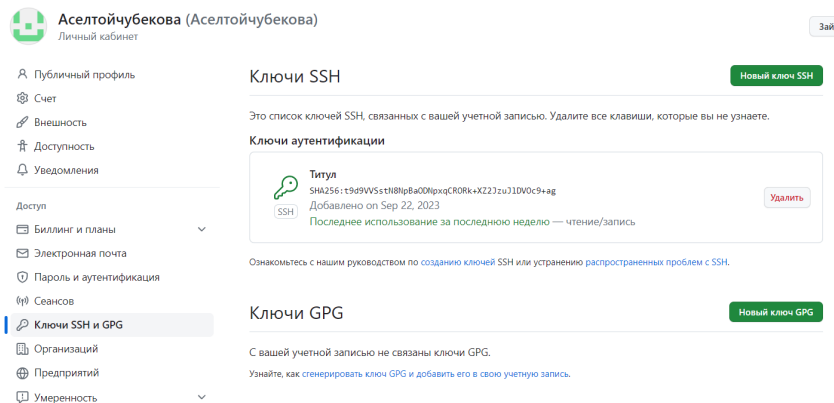


Рис. 2.4: Сгенерированный ключ в github

**Создание рабочего пространства и репозитория курса на основе шаблона.**  
При выполнении лабораторных работ мы должны придерживаться структуры рабочего пространства иерархия которого выглядит так (рис. 2.5):

```
~/work/study/  
└─ 2023-2024/  
    └─ Архитектура компьютера/  
        └─ arch-pc/  
            └─ labs/  
                └─ lab01/  
                    └─ lab02/  
                        └─ lab03/  
                            ...
```

Рис. 2.5: Структура рабочего пространства

Также учтем, что название проекта предмета “Архитектура компьютера” примет следующий вид: study\_2023\_2024\_arch-pc Далее откроем терминал и созда-

дим каталог для предмета “Архитектура компьютера”. Используя команду `cd` и перейдя в этот каталог, убедились, что каталог создан. ((рис. 2.6)) **Создание репозитория курса на основе шаблона.** Создадим репозиторий на основе шаблона, на который мы перейдем по ссылке <https://github.com/yamadharma/course-directory-student-template>. После того как мы перешли по ссылке создадим свой репозиторий на основе шаблона и именем `study_2023– 2024_arh-pc`. Перейдя на свой аккаунт в `github` мы видим, что репозиторий удачно создан, обратим внимание на (рис. 2.7)

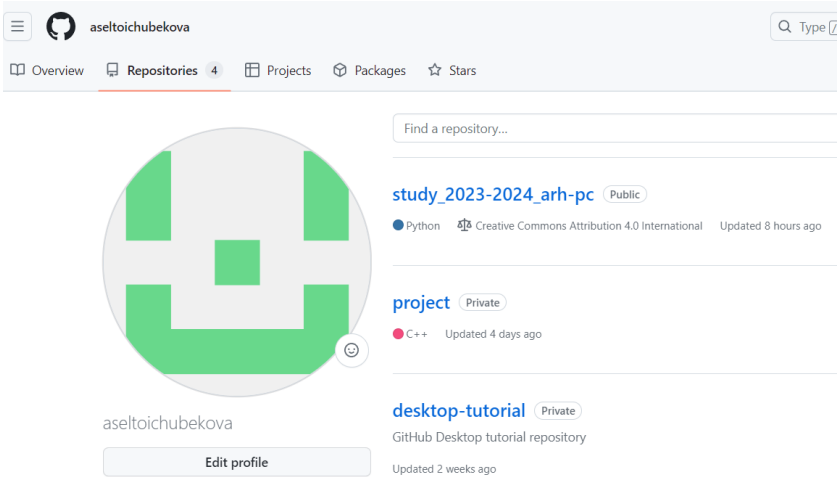


Рис. 2.6: Репозиторий, созданный по шаблону

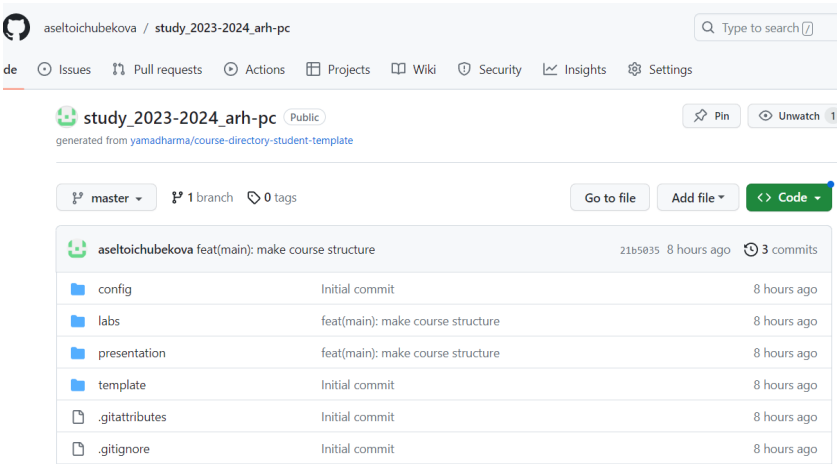


Рис. 2.7: Репозиторий, созданный по шаблону

Далее откроем терминал и перейдем в каталог курса. Затем скопируем, созданный репозиторий. Ссылку для клонирования возьмем на странице созданного репозитория Code -> SSH. Клонирование репозитория необходимо, чтобы начать в нем что-то менять. Введя команды для клонирования далее из результата нашей команды видно, что наш репозиторий клонирован. ((рис. ??))

```
antoychubekova@dk6n55: ~$ mkdir -p ~/work/study/2023-2024/Архитектура компьютера
antoychubekova@dk6n55: ~$ cd ~/work/study/2023-2024/Архитектура компьютера
antoychubekova@dk6n55: ~/work/study/2023-2024/Архитектура компьютера $ git clone --recursive git@github.com:asetoichubekova/study_2023-2024_arch-pc.git arch-pc
Клонирование в «arch-pc»...
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 0), pack-reused 0
Получение объектов: 100% (27/27), 16.91 КБ | 188.80 КБ/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 «./afa/./dk.sci.pfu.edu.ru/home/a/n/antoychubekova/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 0
Получение объектов: 100% (82/82), 92.90 КБ | 1.16 МБ/с, готово.
Определение изменений: 100% (23/23), готово.
Клонирование в «./afa/./dk.sci.pfu.edu.ru/home/a/n/antoychubekova/work/study/2023-2024/Архитектура компьютера/arch-pc/template/report»...
remote: Enumerating objects: 101, done.
remote: Counting objects: 100% (101/101), done.
remote: Compressing objects: 100% (70/70), done.
remote: Total 101 (delta 40), reused 88 (delta 27), pack-reused 0
Получение объектов: 100% (101/101), 327.25 КБ | 2.46 МБ/с, готово.
Определение изменений: 100% (40/40), готово.
Submodule path «template/presentation»: checked out 'b1be380ee91f589264cb755d316174540b753e'
Submodule path «template/report»: checked out '1d1b61dcac9c287a83517b82e3aeff1a33b1e3b2'
```

## Настройка каталога

курса Перейдём в каталог курса в arch-рс. Далее удаляем лишние файлы, которые не хотим отправлять в центральный репозиторий, а точно файл package. Json. Используя команду ls увидели, что файл удален. Вместе с тем создадим необходимый каталог course. Выполняя все команды мы можем видеть результат на экране, (рис. 2.8)

```
antoychubekova@dk6n55: ~/work/study/2023-2024/Архитектура компьютера $ cd arch-pc
antoychubekova@dk6n55: ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ ls
CHANGELOG.md config COURSE LICENSE Makefile package.json README.en.md README.git-flow.md README.md template
antoychubekova@dk6n55: ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ rm package.json
antoychubekova@dk6n55: ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ ls
CHANGELOG.md config COURSE LICENSE Makefile README.en.md README.git-flow.md README.md template
antoychubekova@dk6n55: ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ echo arch-pc > COURSE
antoychubekova@dk6n55: ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ make
```

Рис. 2.8: Удаление ненужного файла. Создание каталога

В конце отправим файлы на сервер. С помощью команды ls или же зайдя в свой аккаунт в github, удостоверились что, создания иерархии рабочего пространства в локальном репозитории и на странице github произвелось правильно. Результат выполненной работы изложен на (рис. 2.9, 2.10, 2.11)

```
antoychubekova@dk6n55: ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ git add .
antoychubekova@dk6n55: ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ git commit -am 'feat(main): make course structure'
[master 195db36] 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
```

Рис. 2.9: Добавление указанных файлов в индекс.

```

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
create mode 100755 labs/lab02/report/pandoc/filters/pandoc_tablenos.py
create mode 100644 labs/lab02/report/pandoc/filters/pandocxnos/__init__.py
create mode 100644 labs/lab02/report/pandoc/filters/pandocxnos/core.py
create mode 100644 labs/lab02/report/pandoc/filters/pandocxnos/main.py
create mode 100644 labs/lab02/report/pandoc/filters/pandocxnos/pandocattributes.py
create mode 100644 labs/lab02/report/report.md
create mode 100644 labs/lab03/presentation/Makefile
create mode 100644 labs/lab03/presentation/image/kulyabov.jpg
create mode 100644 labs/lab03/presentation/presentation.md
create mode 100644 labs/lab03/report/Makefile
create mode 100644 labs/lab03/report/bib/cite.bib
create mode 100644 labs/lab03/report/image/placeimg_800_600_tech.jpg
create mode 100644 labs/lab03/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100755 labs/lab03/report/pandoc/filters/pandoc_eqnos.py
create mode 100755 labs/lab03/report/pandoc/filters/pandoc_fignos.py
create mode 100755 labs/lab03/report/pandoc/filters/pandoc_secnos.py
create mode 100755 labs/lab03/report/pandoc/filters/pandoc_tablenos.py

```

Рис. 2.10: Фиксирование добавленных в индекс изменений.

```

create mode 100755 labs/lab03/report/pandoc/filters/pandoc_tablenos.py
create mode 100644 labs/lab03/report/pandoc/filters/pandocxnos/__init__.py
create mode 100644 labs/lab03/report/pandoc/filters/pandocxnos/core.py
create mode 100644 labs/lab03/report/pandoc/filters/pandocxnos/main.py
create mode 100644 labs/lab03/report/pandoc/filters/pandocxnos/pandocattributes.py
create mode 100644 labs/lab03/report/report.md
create mode 100644 labs/lab04/presentation/Makefile
create mode 100644 labs/lab04/presentation/image/kulyabov.jpg
create mode 100644 labs/lab04/presentation/presentation.md
create mode 100644 labs/lab04/report/Makefile
create mode 100644 labs/lab04/report/bib/cite.bib
create mode 100644 labs/lab04/report/image/placeimg_800_600_tech.jpg
create mode 100644 labs/lab04/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100755 labs/lab04/report/pandoc/filters/pandoc_eqnos.py
create mode 100755 labs/lab04/report/pandoc/filters/pandoc_fignos.py
create mode 100755 labs/lab04/report/pandoc/filters/pandoc_secnos.py
create mode 100755 labs/lab04/report/pandoc/filters/pandoc_tablenos.py
create mode 100644 labs/lab04/report/pandoc/filters/pandocxnos/__init__.py
create mode 100644 labs/lab04/report/pandoc/filters/pandocxnos/core.py
create mode 100644 labs/lab04/report/pandoc/filters/pandocxnos/main.py
create mode 100644 labs/lab04/report/pandoc/filters/pandocxnos/pandocattributes.py
create mode 100644 labs/lab04/report/report.md
create mode 100644 labs/lab05/presentation/Makefile
create mode 100644 labs/lab05/presentation/image/kulyabov.jpg
create mode 100644 labs/lab05/presentation/presentation.md
create mode 100644 labs/lab05/report/Makefile
create mode 100644 labs/lab05/report/bib/cite.bib
create mode 100644 labs/lab05/report/image/placeimg_800_600_tech.jpg
create mode 100644 labs/lab05/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100755 labs/lab05/report/pandoc/filters/pandoc_eqnos.py
create mode 100755 labs/lab05/report/pandoc/filters/pandoc_fignos.py
create mode 100755 labs/lab05/report/pandoc/filters/pandoc_secnos.py
create mode 100755 labs/lab05/report/pandoc/filters/pandoc_tablenos.py
create mode 100644 labs/lab05/report/pandoc/filters/pandocxnos/__init__.py
create mode 100644 labs/lab05/report/pandoc/filters/pandocxnos/core.py
create mode 100644 labs/lab05/report/pandoc/filters/pandocxnos/main.py
create mode 100644 labs/lab05/report/pandoc/filters/pandocxnos/pandocattributes.py
create mode 100644 labs/lab05/report/report.md
create mode 100644 labs/lab06/presentation/Makefile
create mode 100644 labs/lab06/presentation/image/kulyabov.jpg
create mode 100644 labs/lab06/presentation/presentation.md
create mode 100644 labs/lab06/report/Makefile
create mode 100644 labs/lab06/report/bib/cite.bib

```

Рис. 2.11: Фиксирование добавленных в индекс изменений.

```

create mode 100644 presentation/README.md

create mode 100644 presentation/presentation/Makefile
create mode 100644 presentation/presentation/image/kulyabov.jpg
create mode 100644 presentation/presentation/presentation.md
create mode 100644 presentation/report/Makefile
create mode 100644 presentation/report/bib/cite.bib
create mode 100644 presentation/report/image/placeimg_800_600_tech.jpg
create mode 100644 presentation/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100755 presentation/report/pandoc/filters/pandoc_eqnos.py
create mode 100755 presentation/report/pandoc/filters/pandoc_fignos.py
create mode 100755 presentation/report/pandoc/filters/pandoc_secnos.py
create mode 100755 presentation/report/pandoc/filters/pandoc_tablenos.py
create mode 100644 presentation/report/pandoc/filters/pandocxnos/__init__.py
create mode 100644 presentation/report/pandoc/filters/pandocxnos/core.py
create mode 100644 presentation/report/pandoc/filters/pandocxnos/main.py
create mode 100644 presentation/report/pandoc/filters/pandocxnos/pandocattributes.py
create mode 100644 presentation/report/report.md
antoyjchubekova@dk6n55 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ git push
Перечисление объектов: 37, готово.
Подсчет объектов: 100% (37/37), готово.
При сжатии изменений используется до 6 потоков
Сжатие объектов: 100% (29/29), готово.
Запись объектов: 100% (35/35), 342.14 КиБ | 2.63 МБ/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:aseltoichubekova/study_2023-2024_arh-pc.git
   89ab16f..195db36  master -> master
antoyjchubekova@dk6n55 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ ls
CHANGELOG.md  config  COURSE  labs  LICENSE  Makefile  prepare  presentation  README.en.md  README.git-
antoyjchubekova@dk6n55 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ cd labs
antoyjchubekova@dk6n55 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs $ ls
lab01  lab02  lab03  lab04  lab05  lab06  lab07  lab08  lab09  lab10  lab11  README.md  README.ru.md

```

Рис. 2.12: Проверка иерархии в рабочем пространстве

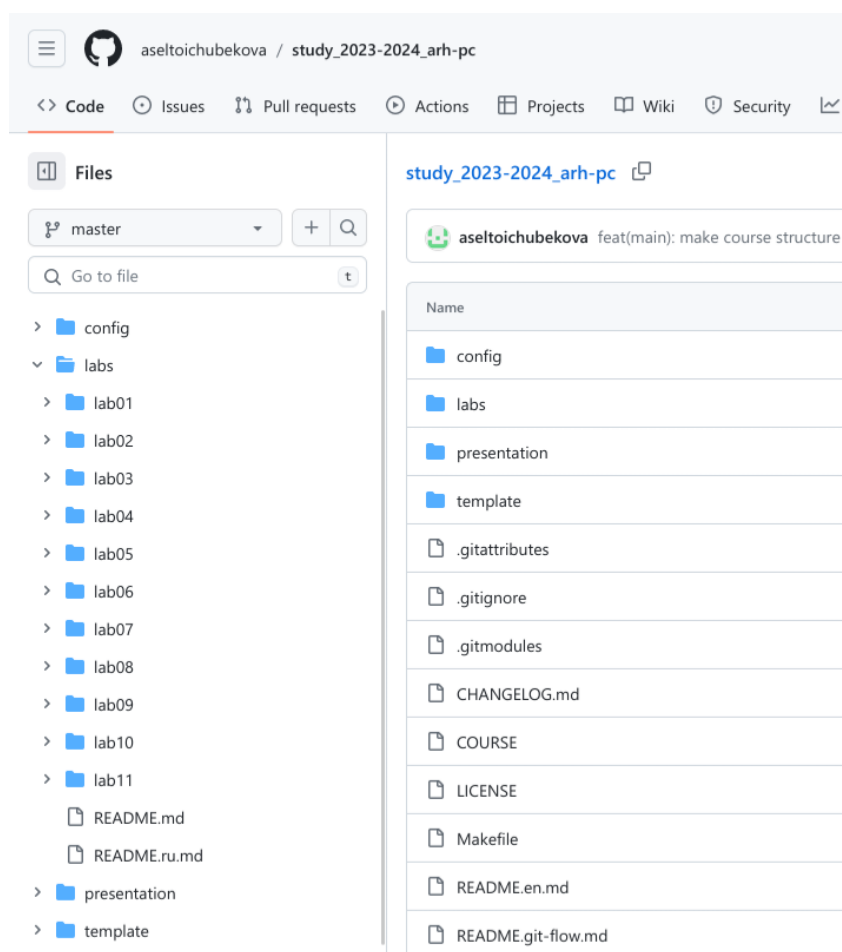


Рис. 2.13: Проверка иерархии в рабочем пространстве

**Задание для самостоятельной работы.** Скопируем отчет по выполненной лабораторной работе №1 в соответствующие каталоги созданного рабочего пространства (labs->lab01->report). Зайдя в свой аккаунт в github, затем перейдя в репозиторий по предмету “Архитектура компьютера”, в указанные каталоги мы видим, что все успешно загрузилось. (рис. 2.14) и (рис. 2.15) Далее, так же загрузим и отчет по проделанной лабораторной работе №2.

```

aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc/labs $ cd lab1
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab1 $ cd report
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab1/report $ cp ~/Документы/Лабораторная работа номер1.pdf ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab1/report
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab1/report $ ls
bib  image  Makefile  pandoc  report.md
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab1/report $ cd ..
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab1 $ cd ..
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc $ git add .
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc $ git commit -m 'feat(main): make course structure'
[feat(main): make course structure]
create mode 100644 labs/lab1/report/Лабораторная работа номер1.pdf
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc $ git push
Подсчет объектов: 10, готово.
Сжатие объектов: 100% (10/10), готово.
Для каких изменений используется до 6 пакетов
Сжатие объектов: 100% (5/5), готово.
Синхронизация: 100% (5/5), 1.28 Мб/с | 5.94 Мб/с, готово.
Всего 6 (изменения 1), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
Синхронизация: 100% (1/1), completed with 3 local objects.
To github.com:aseltouchubekova/study_2023-2024_arh-pc.git
51c0b74..2a45079 master -> master
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc

```

Рис. 2.14: Рис.6.1 Загрузка лабораторной работы №1 в репозиторий.

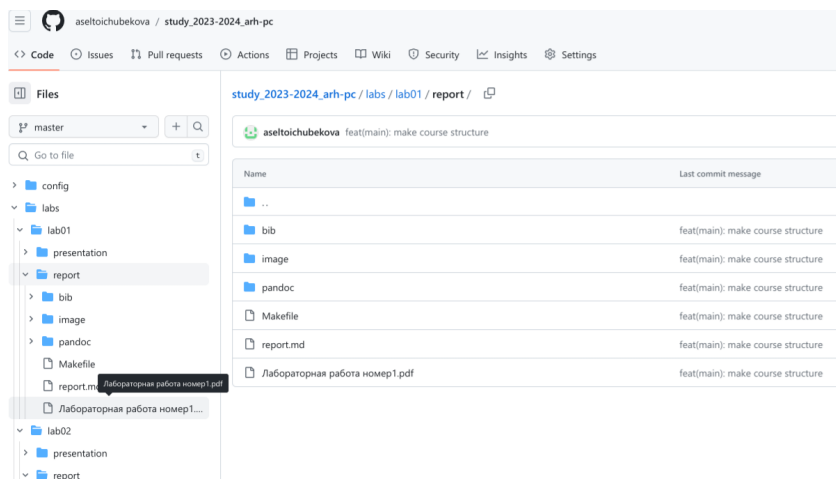


Рис. 2.15: Рис.6.2 Проверка загрузки в Github

Теперь загрузим лабораторную работу номер два в соответствующий каталог(labs->lab02->report). На (рис. 2.16) мы можем просмотреть процесс загрузки, а на (рис. 2.17) проверку в github

```

aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc/labs $ cd lab02
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab02 $ cd report
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab02/report $ cp ~/Документы/Лабораторная работа№2.pdf ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab02/report
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab02/report $ ls
bib  image  Makefile  pandoc  report.md
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab02/report $ cd ..
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab02 $ cd ..
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc $ git add .
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc $ git commit -m 'feat(main): make course structure'
[feat(main): make course structure]
create mode 100644 labs/lab02/report/Лабораторная работа №2.pdf
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc $ git push
Подсчет объектов: 10, готово.
Сжатие объектов: 100% (10/10), готово.
Для каких изменений используется до 6 пакетов
Сжатие объектов: 100% (5/5), готово.
Синхронизация: 100% (5/5), 1.51 Мб/с | 2.28 Мб/с, готово.
Всего 6 (изменения 1), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
Синхронизация: 100% (1/1), completed with 3 local objects.
To github.com:aseltouchubekova/study_2023-2024_arh-pc.git
2a40292..1d21197 master -> master
aseltouchubekova@8: ~$ cd /work/study/2023-2024/Архитектура компьютера/arch-pc

```

Рис. 2.16: Рис.6.3 Загрузка в github лабораторной работы№2



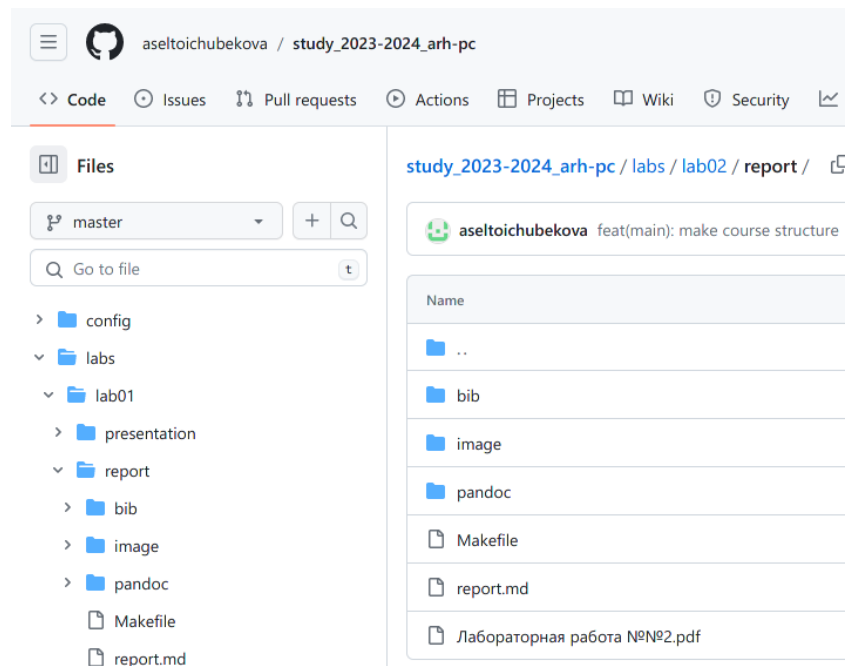


Рис. 2.17: Рис.6.4 Проверка загрузки лабораторной работы№2 в github

### 3 Выводы

В процессе выполнения лабораторной работы №2 я изучила идеологию и применения средств контроля версий, ее функции и разнообразие. Я приобрела практические навыки по работе с одной из популярных систем контроля версии, с системой git. Познакомилась с основными командами git и с web-сервисом github, который требуется для работы с git. Создала рабочее пространство и репозиторий на основе шаблона и SSH-ключи, также научилась работать с каталогами курса, рабочего пространства. А в конце пользуясь приобретенными знаниями загрузила отчет по лабораторной работе №1 и лабораторную работу №2 в соответствующие каталоги, созданного мной репозитория.