

Лабораторная работа №2

Система контроля версии Git

Корчагин Алексей Павлович

Содержание

1	Цель работы	5
2	Теоретическое введение	6
3	Выполнение лабораторной работы	7
4	Выполнение заданий для самостоятельной работы	16
5	Выводы	20
	Список литературы	21

Список иллюстраций

3.1	Созданный аккаунт	7
3.2	Название рисунка	8
3.3	Настройка кодировки	8
3.4	Имя для начальной ветки	8
3.5	Параметр autocrlf	8
3.6	Параметр safecrlf	9
3.7	Ssh ключ	9
3.8	окно SSh and GPG keys	10
3.9	Добавил ключ	10
3.10	создание рабочего пространства	10
3.11	Страница шаблона для репозитория	11
3.12	окно создания репозитория	11
3.13	Созданный	12
3.14	Перемещаюсь между директориями	12
3.15	Клонирование репозитория	13
3.16	Ссылка на клонирование	13
3.17	Удаление файлов	13
3.18	Создание каталогов	14
3.19	Добавление изменений на сервер	14
3.20	Выгрузка изменений на сервер	14
3.21	Страница репозитория	15
4.1	Создание файла	16
4.2	Текстовая работа на отчётом	16
4.3	Перемещаюсь в лаб 1	16
4.4	Проверяю местоположения файлов	17
4.5	Скопировал отчёт в lab1	17
4.6	Добовляю лаб 1 на гит	17
4.7	Сохранил изменения лаб 1	17
4.8	Добавил отчёт по второй лабораторной	18
4.9	Отправка в центральный репозиторий	18
4.10	Каталог lab01/report	18
4.11	Каталог lab012/report	19

Список таблиц

1 Цель работы

Целью работы является изучить идеологию и применение средств контроля версий. Приобрести практические навыки по работе с системой git

2 Теоретическое введение

Git - это набор программ командной строки. Доступ к ним можно получить из терминала посредством ввода команды git с различными опциями. Git является диверсифицированной системой контроля версий. Резервную копию локального поэтому хранилища можно сделать простым копированием или архивацией

Системы контроля версий (Version Control System, VCS) используется 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.

Среди стандартных VCS наиболее известны CVS, Subversion, а среди распределённых — Git, Bazaar, Mercurial. Принципы их работы схожи, отличаются они в основном синтаксисом используемых в работе команд.

Если мы хотим сохранить все изменения в текущем каталоге, то используем:
git add .

Затем сохраняем изменения, поясняя, что было сделано:

git commit -am "Some commit message"

и отправляем в центральный репозиторий:

git push origin имя_ветки или git push

3 Выполнение лабораторной работы

Создал аккаунт на Github (рис. 3.1).

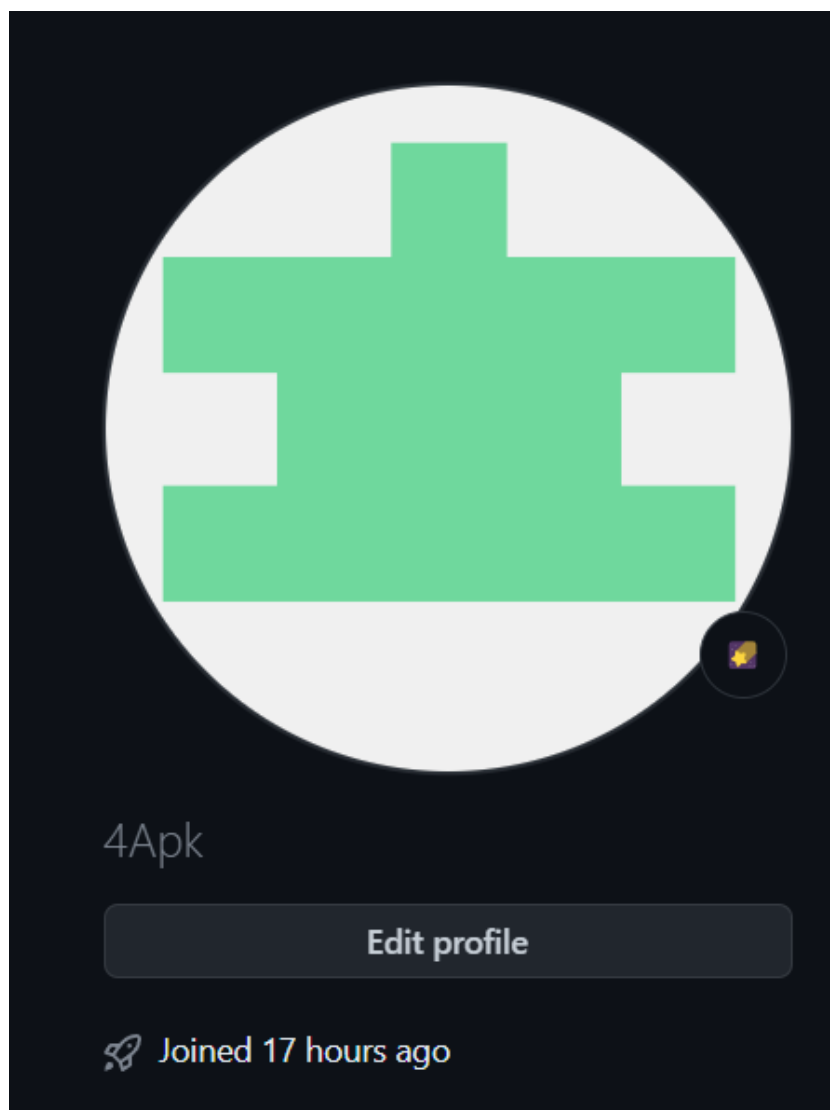


Рис. 3.1: Созданный аккаунт

Открываю виртуальную машину, затем открываю терминал и 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 электронную почту владельца, то есть мою (рис. 3.2).

```
apkorchagin@dk3n64 ~ $ git config --global user.name "<Alexy Korchagin>"
apkorchagin@dk3n64 ~ $ git config --global user.email "<aleshkafpmi@gmail.com>"
apkorchagin@dk3n64 ~ $
```

Рис. 3.2: Название рисунка

Настраиваю utf-8 в выводе сообщений git для корректного отображения символов (рис. 3.3).

```
apkorchagin@dk3n64 ~ $ git config --global core.quotePath false
apkorchagin@dk3n64 ~ $
```

Рис. 3.3: Настройка кодировки

Задаю имя «master» для начальной ветки (рис. 3.4).

```
apkorchagin@dk3n64 ~ $ git config --global init.defaultBranch master
apkorchagin@dk3n64 ~ $
```

Рис. 3.4: Имя для начальной ветки

Задаю параметр `autocrlf` со значением `input`, так как я работаю в системе Linux, чтобы конвертировать CRLF в LF только при коммитах. CR и LF – это символы, которые можно использовать для обозначения разрыва строки в текстовых файлах. (рис. 3.5).

```
apkorchagin@dk3n64 ~ $ git config --global core.autocrlf input
apkorchagin@dk3n64 ~ $
```

Рис. 3.5: Параметр autocrlf

Задаю параметр `safecrlf` со значением `warn`, так Git будет проверять преобразование на обратимость. При значении `warn` Git только выведет предупреждение, но будет принимать необратимые конвертации. (рис. 3.6).

```
apkorchagin@dk3n64 ~ $ git config --global core.safecrlf warn
apkorchagin@dk3n64 ~ $
```

Рис. 3.6: Параметр `safecrlf`

Для последующей идентификации пользователя на сервере репозитория необходимо сгенерировать пару ключей (приватный и открытый). Для этого ввожу команду `ssh-keygen -C "Имя Фамилия, work@email"`, указывая имя владельца и электронную почту владельца. Ключ автоматически сохранится в каталоге `~/.ssh/`. (рис. 3.7).

```
apkorchagin@dk3n64 ~ $ ssh-keygen -C "Alexy Korchagin <aleshkafpmi@gmail.com>"
Generating public/private rsa key pair.
Enter file in which to save the key (/afs/.dk.sci.pfu.edu.ru/home/a/p/apkorchagin/.ssh/id_rsa):
/afs/.dk.sci.pfu.edu.ru/home/a/p/apkorchagin/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /afs/.dk.sci.pfu.edu.ru/home/a/p/apkorchagin/.ssh/id_rsa
Your public key has been saved in /afs/.dk.sci.pfu.edu.ru/home/a/p/apkorchagin/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:S41+fvI1BKdleYlfQeJYu2tfHLVBsjwNnJgXetkb1Ec Alexy Korchagin <aleshkafpmi@gmail.com>
The key's randomart image is:
+---[RSA 3072]-----+
|
|  +==+E|
|  .o=+@o+|
|  .o0==+|
|  o o*o.B|
|  S . . . =|
|  o . . . .|
|  o . . o o|
|  o . +. o |
|  . = . . |
+---[SHA256]-----+
apkorchagin@dk3n64 ~ $
```

Рис. 3.7: Ssh ключ

Открываю браузер, захожу на сайт GitHub. Открываю свой профиль и выбираю страницу «SSH and GPG keys». Нажимаю кнопку «New SSH key» (рис. 3.8).

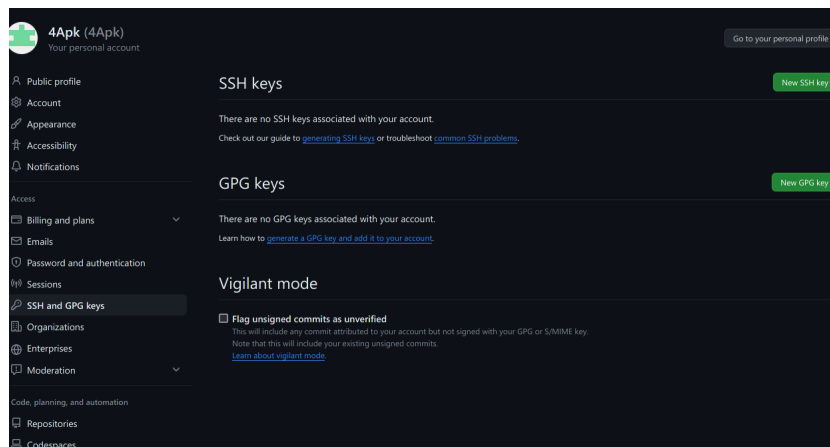


Рис. 3.8: окно SSH and GPG keys

Вставляю скопированный ключ в поле «Key». В поле Title указываю имя для ключа. Нажимаю «Add SSH-key», чтобы завершить добавление ключа (рис. 3.9).

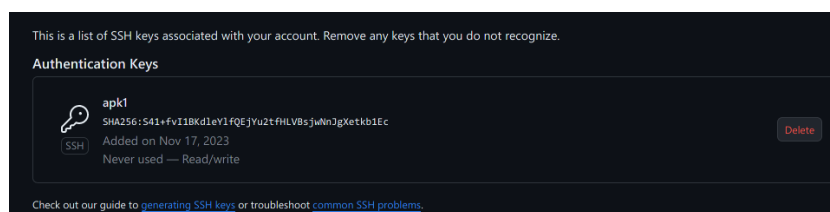


Рис. 3.9: Добавил ключ

Закрываю браузер, открываю терминал. Создаю директорию, рабочее пространство, с помощью утилиты mkdir, благодаря ключу -p создаю все директории после домашней ~/work/study/2023-2024/“Архитектура компьютера” рекурсивно. Далее проверяю с помощью ls, действительно ли были созданы необходимые мне каталоги (рис. 3.10).



Рис. 3.10: создание рабочего пространства

В браузере перехожу на страницу репозитория с шаблоном курса по адресу <https://github.com/yamadharma/course-directory-student-template>. Далее

выбираю «Use this template», чтобы использовать этот шаблон для своего репозитория (рис. 3.11).

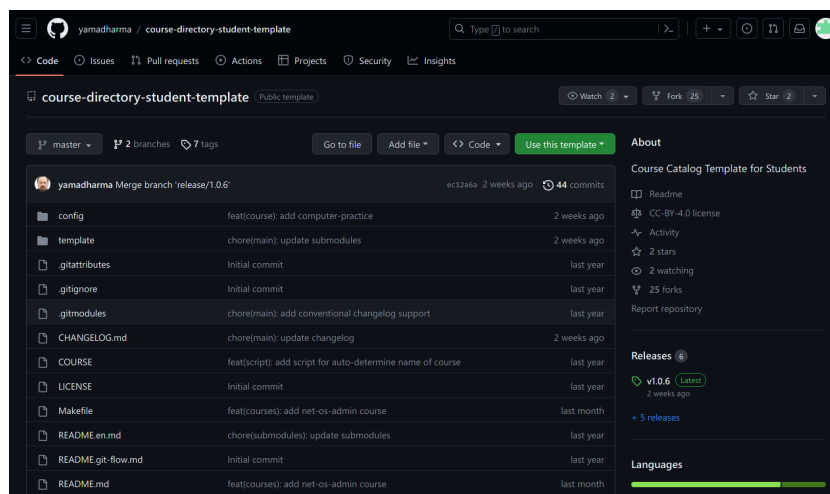


Рис. 3.11: Страница шаблона для репозитория

В открывшемся окне задаю имя репозитория (Repository name): study_2023–2024_arh- pc и создаю репозиторий, нажимаю на кнопку «Create repository from template» (рис. 3.12).

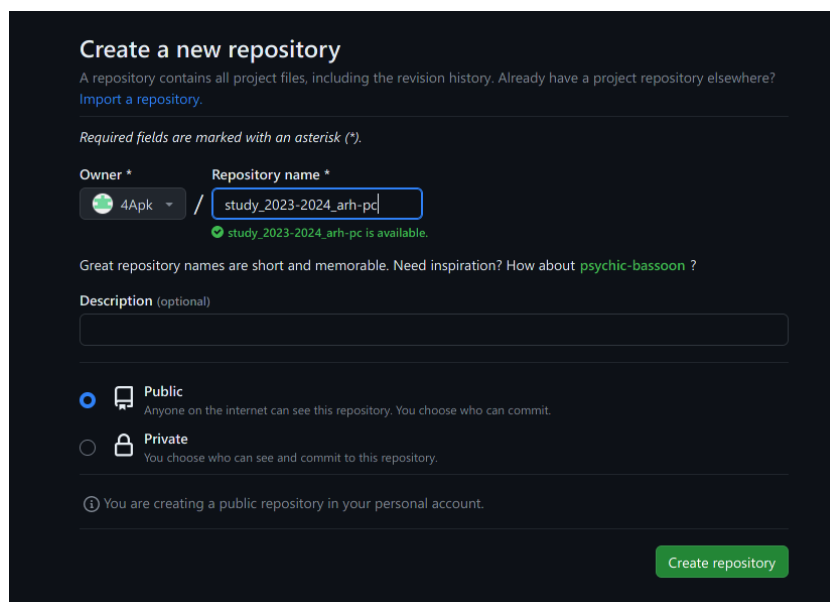


Рис. 3.12: окно создания репозитория

Репозиторий создан (рис. 3.13).

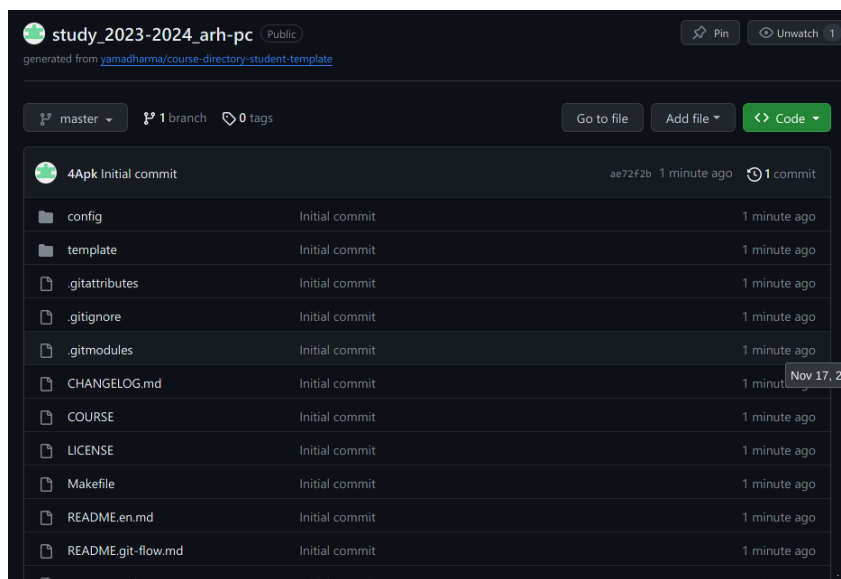


Рис. 3.13: Созданный

Через терминал перехожу в созданный каталог курса с помощью утилиты `cd` (рис. 3.14).

```
apkorchagin@dk3n64 ~ $ cd ~/work/study/2023-2024/"Архитектура компьютера"
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера $
```

Рис. 3.14: Перемещаюсь между директориями

Клонирую созданный репозиторий с помощью команды `git clone -recursive git@github.com:/study_2023-2024_arh-pc.git arch-pc` (рис. 3.15).

```

apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера $ git clone --recursive git@github.com:4Apk/study_2023-2024_arh-pc.git arch-pc
Клонирование в «arch-pc»...
The authenticity of host 'github.com (140.82.121.4)' can't be established.
ED25519 key fingerprint is SHA256:+DIY3wvV6TujHbp2isf/zLDA8zPMSVHdkr4uVCQQu.
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: 38, done.
remote: Counting objects: 100% (38/38), done.
remote: Compressing objects: 100% (29/29), done.
remote: Total 38 (delta 1), reused 17 (delta 0), pack-reused 0
Получение объектов: 100% (38/38), 17.75 KiB | 4.44 MiB/c, готово.
Определение изменений: 100% (1/1), готово.
Подмодуль «template/presentation» (https://github.com/yamadharma/academic-presentation-markdown-template.git) заархивирован по пути «template/presentation»
Подмодуль «template/report» (https://github.com/yamadharma/academic-laboratory-report-template.git) заархивирован по пути «template/report»
Клонирование в «/afs/dk.sci.pfu.edu.ru/home/a/p/apkorchagin/work/study/2023-2024/Архитектура компьютера/arch-pc/template/presentation»...
remote: Enumerating objects: 95, done.
remote: Counting objects: 100% (95/95), done.
remote: Compressing objects: 100% (67/67), done.
remote: Total 95 (delta 34), reused 97 (delta 26), pack-reused 0
Получение объектов: 100% (95/95), 96.99 KiB | 1.05 MiB/c, готово.
Определение изменений: 100% (34/34), готово.
Клонирование в «/afs/dk.sci.pfu.edu.ru/home/a/p/apkorchagin/work/study/2023-2024/Архитектура компьютера/arch-pc/template/report»...
remote: Enumerating objects: 112, done.
remote: Counting objects: 100% (112/112), done.
remote: Compressing objects: 100% (77/77), done.
remote: Total 112 (delta 45), reused 98 (delta 31), pack-reused 0
Получение объектов: 100% (112/112), 331.19 KiB | 2.42 MiB/c, готово.
Определение изменений: 100% (45/45), готово.
Submodule path 'template/presentation': checked out '4ba1761813e197d09e843ff1ca72c69a304f24c'
Submodule path 'template/report': checked out '25e169d367953f60c76c251db299ed52852b401f'
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера $

```

Рис. 3.15: Клонирование репозитория

Копирую ссылку для клонирования на странице созданного репозитория, сначала перейдя в окно «code», далее выбрав в окне вкладку “SSH” (рис. 3.16).

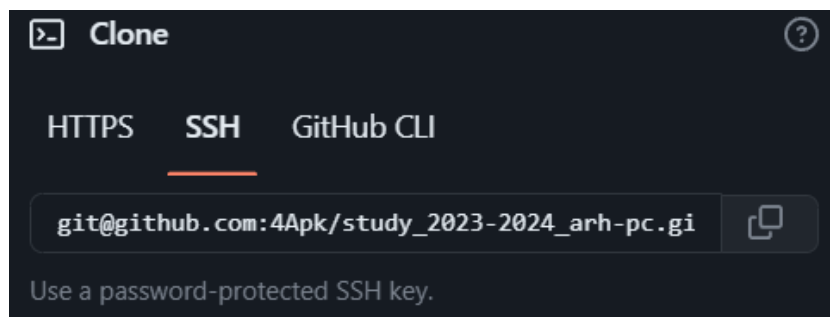


Рис. 3.16: Ссылка на клонирование

Перехожу в каталог arch-pc с помощью утилиты cd и удаляю лишние файлы при помощи rm (рис. 3.17).

```

apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера $ cd ~/work/study/2023-2024/Архитектура компьютера/arch-pc
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ rm package.json
bash: rm.package.json: команда не найдена
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ rm package.json

```

Рис. 3.17: Удаление файлов

Создаю необходимые каталоги (рис. 3.18).

```

apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ echo arch-pc > COURSE
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ make
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $

```

Рис. 3.18: Создание каталогов

Отправляю созданные каталоги с локального репозитория на сервер: добавляю все созданные каталоги с помощью `git add`, комментирую и сохраняю изменения на сервере как добавление курса с помощью `git commit` (рис. 3.19).

```

apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ git add .
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ git commit -am 'feat(main): make course structure'
[master 455fcb7] 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

```

Рис. 3.19: Добавление изменений на сервер

Отправляю все на сервер с помощью `push` (рис. 3.20).

```

apkorchagin@dk3n64 ~/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 | 9.00 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:4Apk/study_2023-2024_arh-pc.git
ae72f2b..455fcb7 master -> master

```

Рис. 3.20: Выгрузка изменений на сервер

Проверяю корректность выполнения работы сначала на самом сайте GitHub (рис. 3.21).

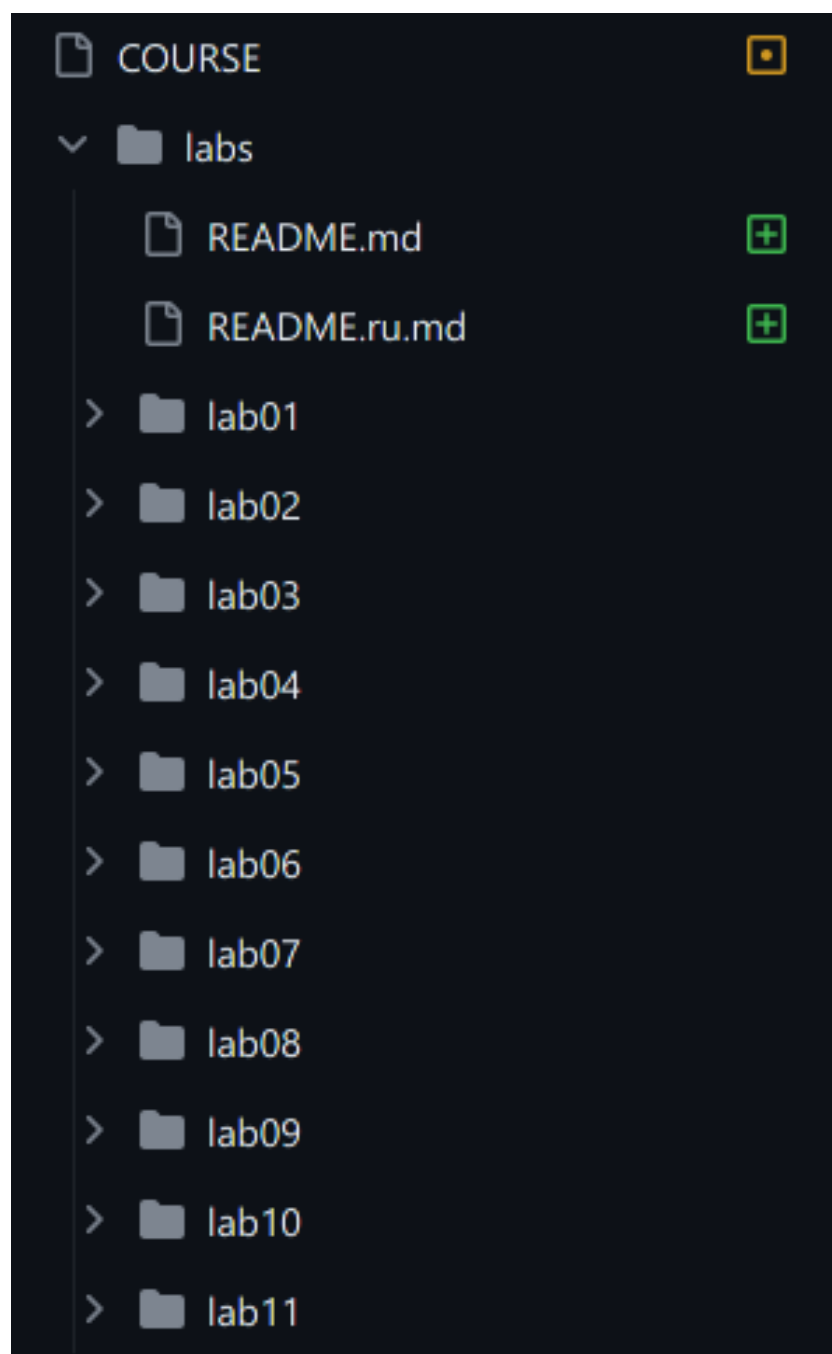


Рис. 3.21: Страница репозитория

4 Выполнение заданий для самостоятельной работы

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. 4.1).

```
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab02/report $ touch л02_Корчагин_отчет
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab02/report $
```

Рис. 4.1: Создание файла

Оформить отчет я смогу в текстовом процессоре LibreOffice Writer (рис. 4.2).

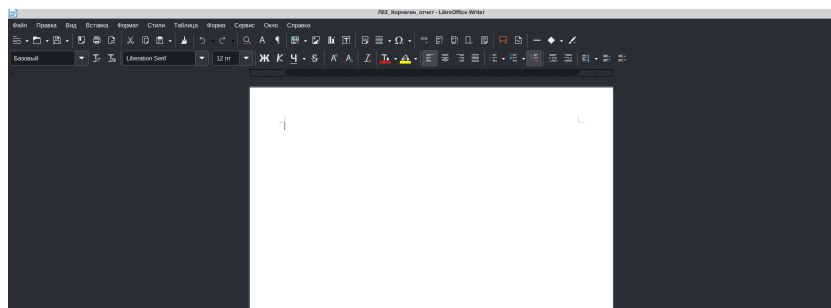


Рис. 4.2: Текстовая работа на отчётом

Перехожу из подкаталога `lab02/report` в подкаталог `lab01/report` с помощью утилиты `cd` (рис. 4.3).

```
apkorchagin@dk3n64 ~ $ cd ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab01/report/
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab01/report $
```

Рис. 4.3: Перемещаюсь в лаб 1

Проверяю местонахождение файла с отчетом по первой лабораторной работе. Он должен быть в подкаталоге домашней директории «Загрузки», для проверки использую команду `ls` (рис. 4.4).

```
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab01/report $ ls ~/Загрузки
'отчет_лаб_1, Корчагин А.П. НММ-02-23.doc'
```

Рис. 4.4: Проверяю местоположения файлов

Копирую первую лабораторную с помощью утилиты `sr` и проверяю корректность выполнения команды `sr` с помощью `ls` (рис. 4.5).

```
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab01/report $ ls
bib image Makefile pandoc report.md Л01_Корчагин_отчет.doc
```

Рис. 4.5: Скопировал отчёт в lab1

Добавляю с помощью команды `git add` в коммит созданные файлы: `Л01_Корчагин_отчет` (рис. 4.6).

```
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab01/report $ git add Л01_Корчагин_отчет.doc
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab01/report $
```

Рис. 4.6: Добавляю лаб 1 на гит

Сохраняю изменения на сервере командой `git commit -m "..."`, поясняя, что добавила файлы (рис. 4.7).

```
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab01/report $ git commit -m "Add existing Files"
[master 63ea825] Add existing Files
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100755 labs/lab01/report/Л01_Корчагин_отчет.doc
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab01/report $
```

Рис. 4.7: Сохранил изменения лаб 1

То же самое делаю для отчета по второй лабораторной работе: перехожу в директорию `labs/lab02/report` с помощью `cd`, добавляю с помощью `git add` нужный файл, сохраняю изменения с помощью `git commit` (рис. 4.8).

```

apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab02/report $ git add Л02_Корчагин_отчет.doc
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab02/report $ git commit -m "add existing file"
[master 5ab8c01] add existing file
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 labs/lab02/report/Л02_Корчагин_отчет.doc
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab02/report $

```

Рис. 4.8: Добавил отчёт по второй лабораторной

Отправляю в центральный репозиторий сохраненные изменения командой `git push -f origin master` (рис. 4.9).

```

apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab02/report $ git push -f origin master
Перечисление объектов: 17, готово.
Подсчет объектов: 100% (15/15), готово.
При сжатии изменений используется до 6 потоков
Сжатие объектов: 100% (11/11), готово.
Запись объектов: 100% (11/11), 525.09 Киб | 1.91 Миб/с, готово.
Всего 11 (изменений 5), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (5/5), completed with 2 local objects.
To github.com:4Apk/study_2023-2024_arh-pc.git
 455fcb7..5ab8c01 master -> master
apkorchagin@dk3n64 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab02/report $

```

Рис. 4.9: Отправка в центральный репозиторий

При просмотре изменений на Github вижу, что были добавлены файлы с отчетами по лабораторным работам (рис. 4.10).

4Apk Add existing Files 63ea025	
Name	Last commit message
..	
bib	feat(main): make course structure
image	feat(main): make course structure
pandoc	feat(main): make course structure
Makefile	feat(main): make course structure
report.md	feat(main): make course structure
ЛЮ1_Корчагин_отчет.doc	Add existing Files

Рис. 4.10: Каталог lab01/report

(рис. 4.11).

4Apk add existing file 5ab8c01	
Name	Last commit message
..	
bib	feat(main): make course structure
image	feat(main): make course structure
pandoc	feat(main): make course structure
Makefile	feat(main): make course structure
report.md	feat(main): make course structure
Л02_Корчагин_отчет.doc	add existing file

Рис. 4.11: Каталог lab012/report

5 Выводы

При выполнении данной лабораторной работы я изучил идеи и применение средств контроля версий, а также приобрела практические навыки по работе с системой git.

Список литературы