

отчёт по лабораторной работе №2

Дисциплина: Архитектура компьютера

Курилко-Рюмин Евгений Михайлович

Содержание

1	Цель работы	4
2	Задание	5
3	Теоретическое введение	6
4	Выполнение лабораторной работы	8
5	Выводы	22
	Список литературы	23

Список иллюстраций

4.1	Создание учетной записи GitHub	8
4.2	Предварительная конфигурация git	9
4.3	Создание имени для начальной ветки, присваивание параметров	9
4.4	Генерация SSH-ключа	10
4.5	Установка утилиты xclip	10
4.6	Добавление ключа	11
4.7	Создание рабочего пространства	12
4.8	Страница репозитория с шаблоном	12
4.9	Созданный репозиторий	13
4.10	Клонирование репозитория	14
4.11	Окно с ссылкой на копирование репозитория	15
4.12	Добавление и сохранение изменений на сервере	16
4.13	Выгрузка изменений на сервер	16
4.14	Проверка результата на странице репозитория	17
4.15	Перемещение по директориям, создание файла	17
4.16	Работа с отчётом в текстовом редакторе	18
4.17	Перемещение между директориями, проверка местонахождения файла	18
4.18	Перемещение между директориями, копирование файла	18
4.19	Добавление файла на сервер	19
4.20	Отправка в центральный репозиторий сохраненных изменений .	19
4.21	Отправка в центральный репозиторий сохраненных изменений .	19
4.22	Страница последних изменений в репозитории	20
4.23	Проверка каталогов	21

1 Цель работы

Целью данной работы является приобретение практического опыта работы с системой git, изучение принципов и применения контроля версий.

2 Задание

1. Настройка GitHub.
2. Базовая настройка Git.
3. Создание SSH-ключа.
4. Создание рабочего пространства.
5. Создание репозитория курса на основе шаблона.
6. Настройка каталога курса.
7. Выполнение задания для самостоятельной работы.

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 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 поддерживают возможность отслеживания и разрешения конфликтов, которые могут возникнуть при работе нескольких человек над одним файлом. Можно объединить изменения, сделанные разными участниками, вручную выбрать нужную версию, отменить изменения вовсе или заблокировать файлы для изменения. В зависимости от настроек блокировка не позволяет

другим пользователям получить рабочую копию или препятствует изменению рабочей копии файла средствами файловой системы ОС, обеспечивая таким образом привилегированный доступ только одному пользователю, работающему с файлом. 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. 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. Среди классических VCS наиболее известны CVS, Subversion, а среди распределённых — Git, Bazaar, Mercurial. Принципы их работы схожи, отличаются они в основном синтаксисом используемых в работе команд. 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.

4 Выполнение лабораторной работы

1. Настройка GitHub

Создаю учетную запись на сайте GitHub, ввожу нужные данные учетной записи (рис.1)

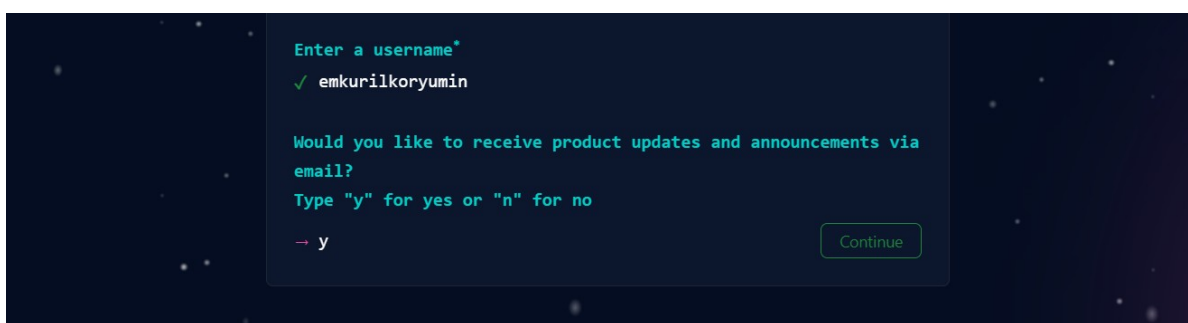
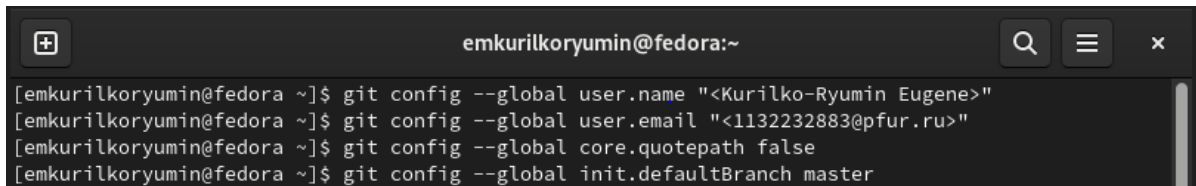


Рис. 4.1: Создание учетной записи GitHub

2. Базовая настройка Git

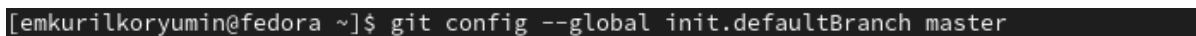
Открываю виртуальную машину, затем запускаю терминал, делаю предварительную конфигурацию git. Ввожу команду `git config --global user.name`, указывая свое имя и команду `git config --global user.email «work@mail»`, указывая в этой команде свою электронную почту, настраиваю utf-8 в выводе сообщений git для корректного отображения символов (рис. 2)

A terminal window titled 'emkurilkoryumin@fedora:~' with search, menu, and close icons in the top right. It contains four lines of terminal output:

```
[emkurilkoryumin@fedora ~]$ git config --global user.name "<Kurilko-Ryumin Eugene>"
[emkurilkoryumin@fedora ~]$ git config --global user.email "<1132232883@pfur.ru>"
[emkurilkoryumin@fedora ~]$ git config --global core.quotePath false
[emkurilkoryumin@fedora ~]$ git config --global init.defaultBranch master
```

Рис. 4.2: Предварительная конфигурация git

Задаю имя начальной ветке, а также параметры `autocrlf` и `safecrlf`, причем параметр `autocrlf` дополняем значением `input`, для конвертации символов разрыва строки в текстовых файлах (CRLF и LF) только при коммитах. Параметру `safecrlf` задаю значение `warn`, так Git будет проверять преобразование на обратимость, и при данном значении будет выведено только предупреждение, а необратимые конвертации будут приняты (рис. 3)

A terminal window showing a single line of terminal output:

```
[emkurilkoryumin@fedora ~]$ git config --global init.defaultBranch master
```

Рис. 4.3: Создание имени для начальной ветки, присваивание параметров

3. Создание SSH-ключа

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 своё имя и электронную почту. Ключ автоматически сохранится в каталоге `~/.ssh/`. (рис. 4)

```

[emkurilkoryumin@fedora ~]$ ssh-keygen -C "Kurilko-Ryumin Eugene 1132232883@pfur.ru"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/emkurilkoryumin/.ssh/id_rsa):
Created directory '/home/emkurilkoryumin/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/emkurilkoryumin/.ssh/id_rsa
Your public key has been saved in /home/emkurilkoryumin/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:ejJdMvsm+JKSCEcFH0o45VzwqRX4RSjz4wHZD/eBhG4 Kurilko-Ryumin Eugene 1132232883@pfur.ru>
The key's randomart image is:
+---[RSA 3072]-----+
|  O=*=+O.          |
| O+*=+++ .         |
| .+O= . .          |
| .oE . .           |
| ..O O S .         |
| . . . O =         |
| O . .+O+          |
| . O ++...         |
| . O.O.            |
+---[SHA256]-----+
[emkurilkoryumin@fedora ~]$

```

Рис. 4.4: Генерация SSH-ключа

Устанавливаю утилиту `xclip`, позволяющую копировать любой текст через терминал. Использую команду «`dnf install`» с ключом `-u` от имени суперпользователя, ввожу в начале «`sudo`» (рис. 5)

```

emkurilkoryumin@fedora:~$ sudo dnf install xclip
RPM Fusion for Fedora 38 - Nonfree - NVIDIA Driver 26 kB/s | 14 kB 00:00
RPM Fusion for Fedora 38 - Nonfree - Steam 5.5 kB/s | 2.2 kB 00:00
Загрузки: 26 kB/s | 14 kB 00:00
Установки: 5.5 kB/s | 2.2 kB 00:00
-----
Пакет Архитектура Версия Репозиторий Размер
-----
Установка: xclip x86_64 0.13-19.git11c8a61.fc38 fedora 37 k
-----
Результат транзакции
-----
Установка 1 пакет
-----
Объем загрузки: 37 k
Объем изменений: 63 k
Загрузка пакетов:
xclip-0.13-19.git11c8a61.fc38.x86_64.rpm 1.1 MB/s | 37 kB 00:00
-----
Общий размер: 52 kB/s | 27 kB 00:00
Fedora 38 - x86_64 1.6 MB/s | 1.6 kB 00:00
Идентификатор GPG-ключа: 8E8108464:
Идентификатор поставщика: "Fedora (38) <fedora.38.primary@fedoraproject.org>"
Опекатор: 6451 8048 842D 5467 8017 1221 8064 807C ED19 3464
Источник: /etc/pki/gnupg/gpg/RPM-GPG-KEY-Fedora-38-x86_64
Импорт ключа успешно завершен
Проверка транзакции
Проверка транзакции успешно завершена.
Идет проверка транзакции
Лист транзакции: проведен успешно.
Выполнение транзакции
Подготовка :
Установка : xclip-0.13-19.git11c8a61.fc38.x86_64 1/1
Запуск скриптов: xclip-0.13-19.git11c8a61.fc38.x86_64 1/1
Проверка : xclip-0.13-19.git11c8a61.fc38.x86_64 1/1
-----
Установлен:
xclip-0.13-19.git11c8a61.fc38.x86_64
Выполнено:
[emkurilkoryumin@fedora ~]$ cat
2
1): Остановлен cat
[emkurilkoryumin@fedora ~]$ cat ~/.ssh/id_rsa.pub | xclip -sel clip
[emkurilkoryumin@fedora ~]$

```

Рис. 4.5: Установка утилиты `xclip`

Открываю браузер, захожу на сайт GitHub. Открываю свой профиль и выбираю страницу «SSH and GPG keys». Нажимаю кнопку «New SSH key». Вставляю скопированный ключ в поле «Key». В поле Title указываю имя для ключа. Нажимаю «Add SSH-key», чтобы завершить добавление ключа (рис. 6)

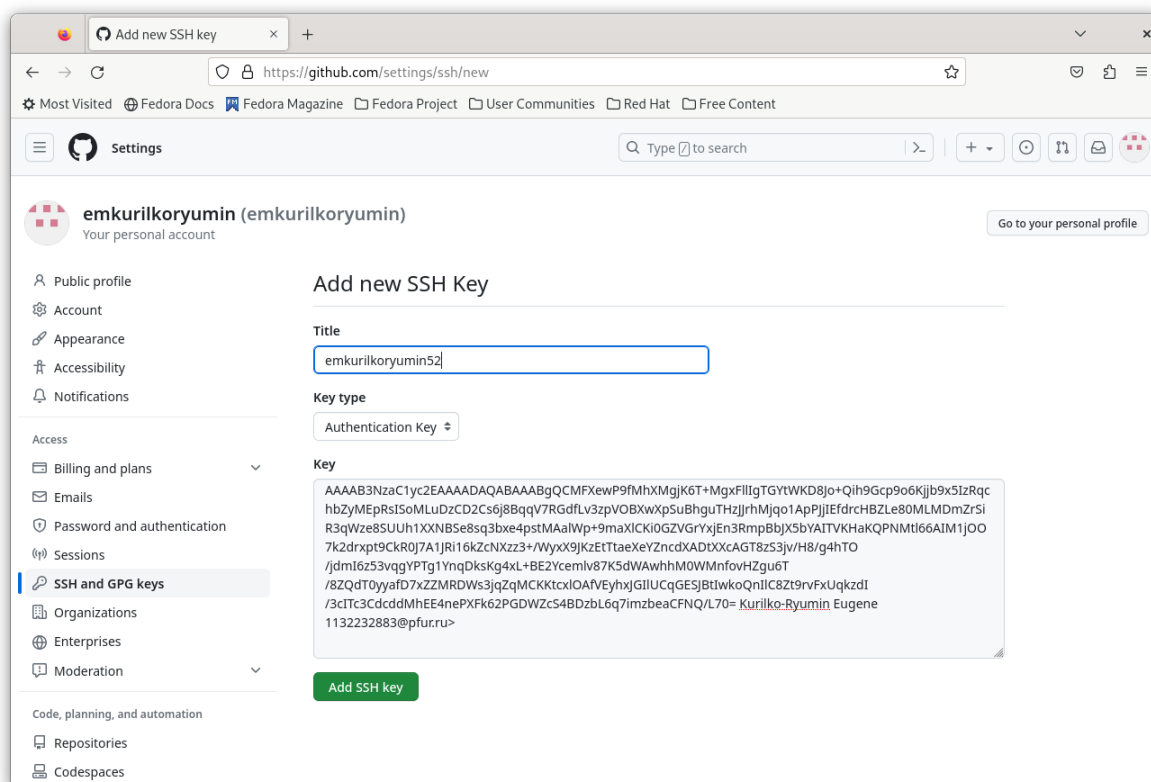


Рис. 4.6: Добавление ключа

4. Создание рабочего пространства

Закрываю браузер, открываю терминал. Создаю директорию, рабочее пространство с помощью утилиты mkdir. С помощью ключа -p создаю рекурсивно все директории после домашней ~/work/study/2023-2024/“Архитектура Компьютера”. Далее проверяю с помощью ls, действительно ли были созданы необходимые мне каталоги (рис. 7)

```

[emkurilkoryumin@fedora ~]$ mkdir -p work/study/2023-2024/"Архитектура Компьютеров"
[emkurilkoryumin@fedora ~]$ ls
work  Видео  Документы  Загрузки  Изображения  Музыка  Общедоступные  'Рабочий стол'  Шаблоны
[emkurilkoryumin@fedora ~]$

```

Рис. 4.7: Создание рабочего пространства

5. Создание репозитория курса на основе шаблона

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. 8)

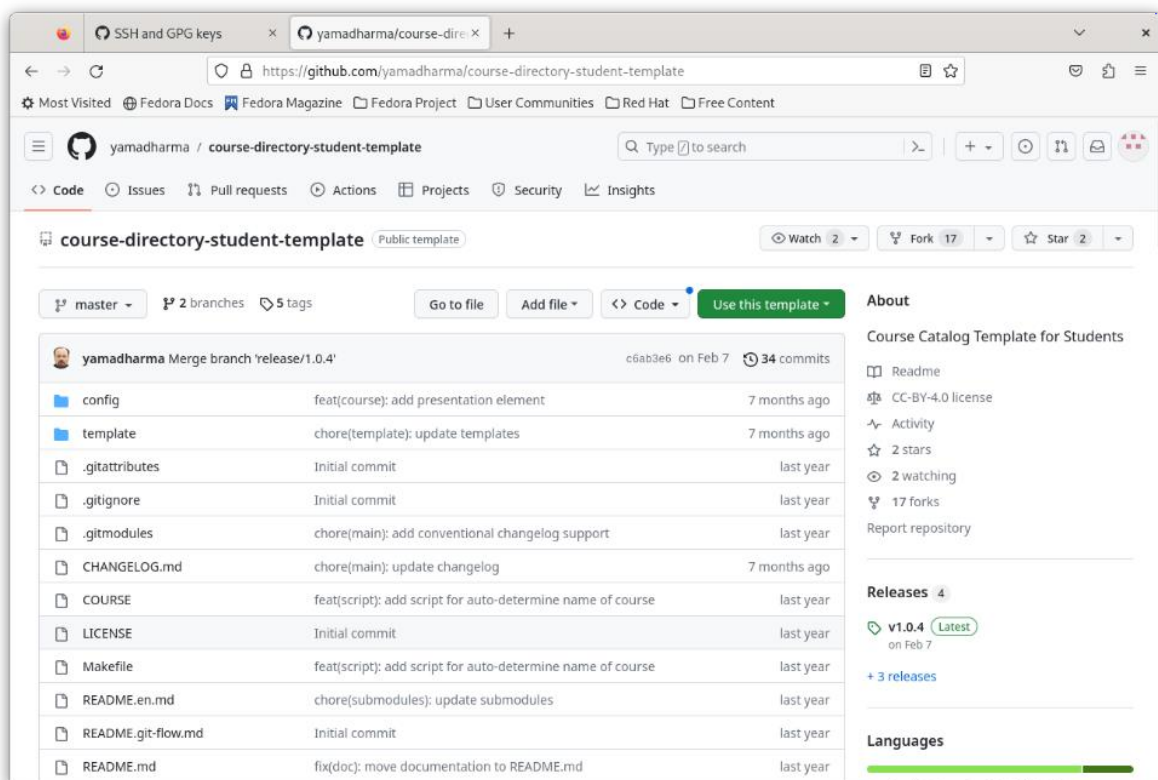


Рис. 4.8: Страница репозитория с шаблоном

Ждем создания репозитория, затем он откроется (рис. 9)

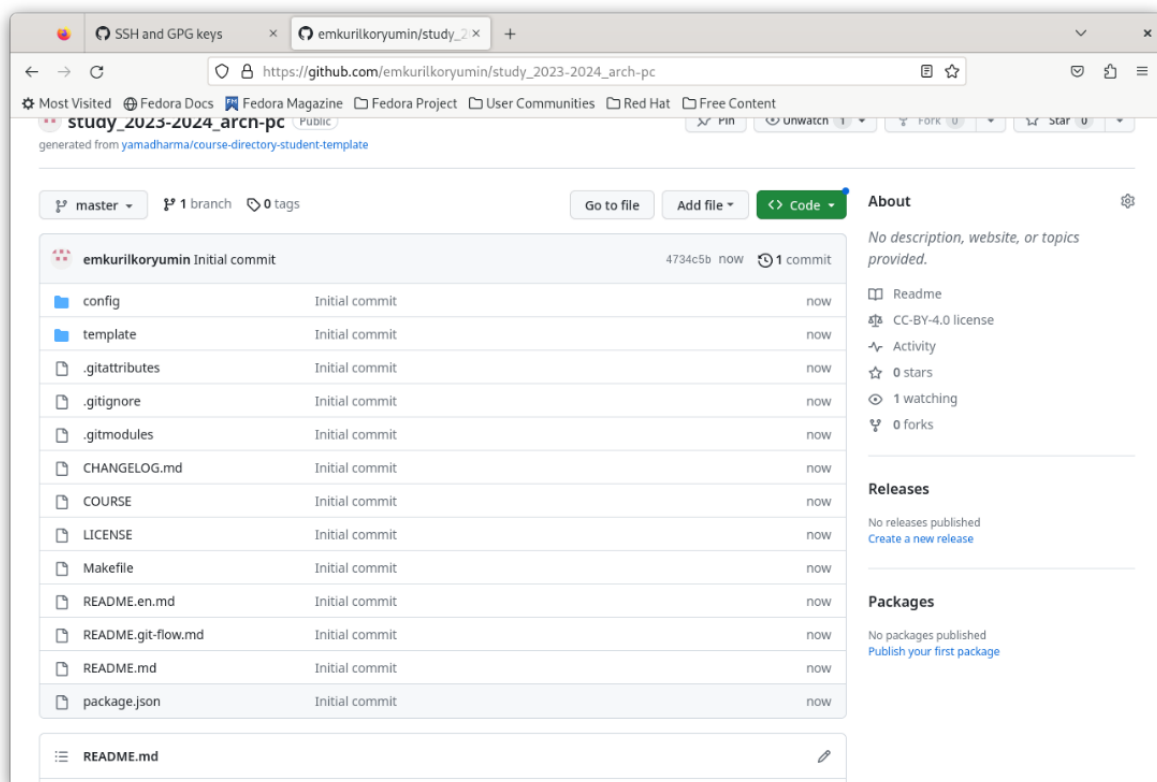


Рис. 4.9: Созданный репозиторий

Далее через терминал перехожу в созданный каталог курса с помощью утилиты `cd`. Клонировать созданный репозиторий с помощью команды: «`git clone –recursive git@github.com:/study_2023–2024_arh-pc.git arch-pc`» (рис. 10)

```

[emkuril@koryuminfedora ~]$ mkdir -p work/study/2023-2024/«Архитектура Компьютеров»
[emkuril@koryuminfedora ~]$ ls
work  Работы  Документы  Зарплата  Исследования  Музыка  Стандартные  «Рабочий стол»  Настройка
[emkuril@koryuminfedora ~]$ cd
[emkuril@koryuminfedora ~]$ cd ~/work/study/2023-2024/«Архитектура Компьютеров»
[emkuril@koryuminfedora Архитектура Компьютеров]$ git clone --recursive git@github.com:emkuril@koryumin/study_2023-2024_arch-pc.git arch-pc
Клонирование в «arch-pc»...
sh: Could not resolve hostname github.com: Name or service not known
fatal: Не удалось прочесть из внешнего репозитория.

Удостоверьтесь, что у вас есть необходимые права доступа
к репозиторию, существует.
[emkuril@koryuminfedora Архитектура Компьютеров]$ git clone --recursive git@github.com:emkuril@koryumin/study_2023-2024_arch-pc.git arch-pc
Клонирование в «arch-pc»...
The authenticity of host 'github.com (140.82.121.3)' can't be established.
ED25519 key fingerprint is SHA256:D1V3wvV6TuJ3hbp2isF/zLDA0ZPMSvdkr4UvCoQd.
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: 27, done.
remote: Counting objects: 100% (27/27), done.
remote: Compressing objects: 100% (20/20), done.
remote: Total 27 (delta 1), reused 11 (delta 0), pack-reused 0
Получение объектов: 100% (27/27), 16.93 KiB | 16.93 MiB/c, 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/emkuril@koryumin/work/study/2023-2024/Архитектура Компьютеров/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 KiB | 980.00 KiB/c, готово.
Определение изменений: 100% (28/28), готово.
Клонирование в «/home/emkuril@koryumin/work/study/2023-2024/Архитектура Компьютеров/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 KiB | 1.97 MiB/c, готово.
Определение изменений: 100% (40/40), готово.
Submodule path «template/presentation»: checked out 'b1be3809ee91f5809264cb755d316174540b753e'
Submodule path «template/report»: checked out '1d1b61dcac9c287a83917b82e3ae71a33b1e3b2'
[emkuril@koryuminfedora Архитектура Компьютеров]$

```

Рис. 4.10: Клонирование репозитория

Ссылку для копирования можно взять на странице созданного репозитория, сначала перейдя в окно «code», далее выбрав в окне вкладку «SSH» (рис. 11)

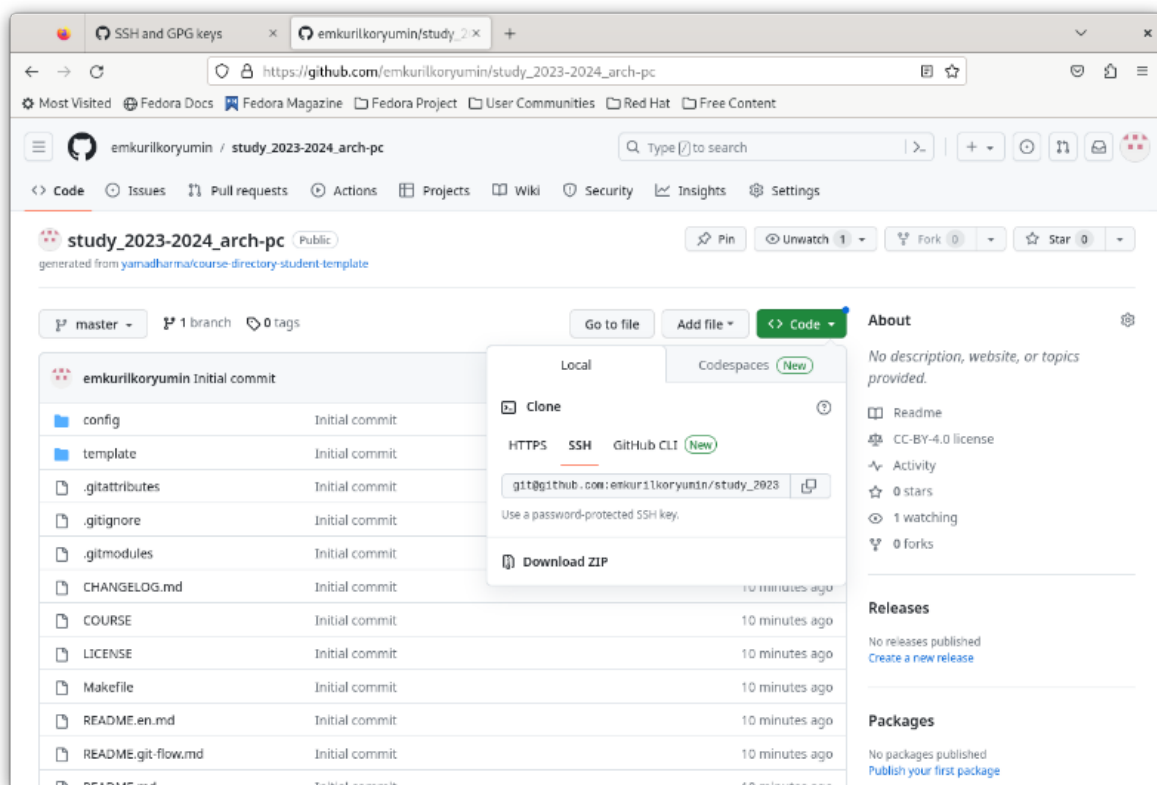


Рис. 4.11: Окно с ссылкой на копирование репозитория

6. Настройка каталога курса

Отправляю созданные каталоги с локального репозитория на сервер: добавляю все созданные каталоги с помощью «git add», комментирую и сохраняю изменения на сервере с помощью «git commit» (рис. 12)

```
emkurlikoryumin@fedora:~/work/study/2023-2024/Архитектура Компьютеров/arch-pc
[emkurlikoryumin@fedora arch-pc]$ git add
[emkurlikoryumin@fedora arch-pc]$ git commit -m 'feat(main): make course structure'
master b9d1717 feat(main): make course structure
199 files changed, 24725 insertions(+), 14 deletions(-)
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/placement_800_600_tech.jpg
create mode 100644 labs/lab01/report/pandoc/csl/gost-r-7-9-5-2008-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/placement_800_600_tech.jpg
create mode 100644 labs/lab02/report/pandoc/csl/gost-r-7-9-5-2008-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/placement_800_600_tech.jpg
create mode 100644 labs/lab03/report/pandoc/csl/gost-r-7-9-5-2008-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
```

Рис. 4.12: Добавление и сохранение изменений на сервере

Отправляю все на сервер с помощью «push» (рис. 13)

```
[emkurlikoryumin@fedora arch-pc]$ git push
Перечисление объектов: 37, готово.
Подсчет объектов: 100% (37/37), готово.
Сжатие объектов: 100% (35/35), 342.14 Кб | 2.57 Мб/с, готово.
Запись объектов: 100% (35/35), 342.14 Кб | 2.57 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:emkurlikoryumin/study.2023-2024_arch-pc.git
 473dc5b..b9d1717 master -> master
[emkurlikoryumin@fedora arch-pc]$
```

Рис. 4.13: Выгрузка изменений на сервер

Проверяю правильность выполнения работы на самом сайте GitHub (рис.14)

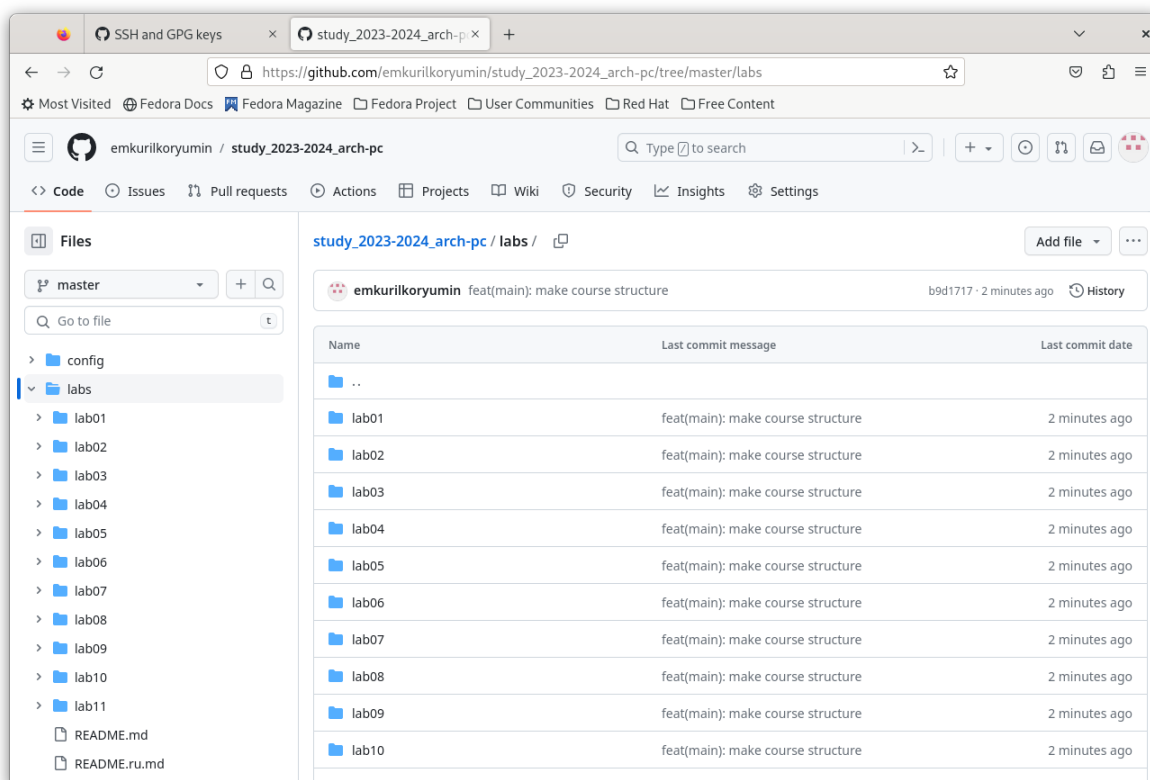


Рис. 4.14: Проверка результата на странице репозитория

7. Выполнение заданий для самостоятельной работы

Перехожу в директорию `labs/lab02/report` с помощью утилиты `cd`. Создаю в каталоге файл для отчета по третьей лабораторной работе с помощью утилиты `touch`. (рис. 15)

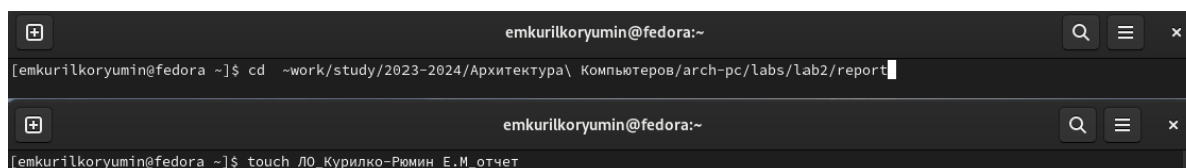


Рис. 4.15: Перемещение по директориям, создание файла

- 1) Оформить отчет я смогу в текстовом процессоре LibreOffice Writer, найдя его в меню приложений. После открытия текстового процессора открываю в нем созданный файл. Теперь можно создать в нем отчет (рис.16)

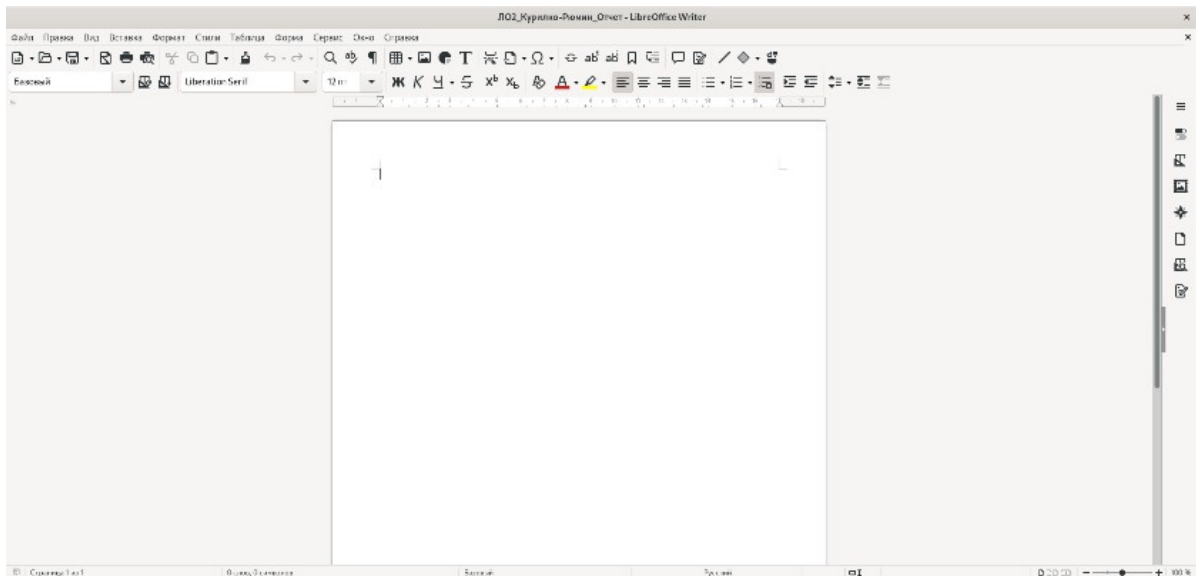


Рис. 4.16: Работа с отчётом в текстовом редакторе

- 2) Проверяю местонахождение файла с отчетом по первой лабораторной работе, используя команду ls (рис. 17)

```
[emkurilkoryumin@fedora report]$ ls ~/Загрузки
Л02_Курилко-Рюмин_Отчет  'ЛОМ1_Курилко-Рюмин Е.М_отчет (1).pdf'
[emkurilkoryumin@fedora report]$
```

Рис. 4.17: Перемещение между директориями, проверка местонахождения файла

Перехожу из подкаталога lab01/report в подкаталог lab02/report с помощью утилиты cd. Копирую вторую лабораторную с помощью утилиты cp. (рис. 18)

```
[emkurilkoryumin@fedora report]$ cp ~/Загрузки/Л02_Курилко-Рюмин Е.М_отчет home/work/study/2023-2024//Архитектура\ Компьютеров/arch-pc/labs/lab02/report
```

Рис. 4.18: Перемещение между директориями, копирование файла

- 3) Добавляю с помощью команды git add в коммит созданные файлы:

ЛОН№1_Курилко-Рюмин Е.М_отчет. Перехожу в директорию, в которой находится отчет по второй лабораторной. Добавляю файл ЛОН№1_Курилко-Рюмин Е.М_отчет (рис. 20)

```
[emkurilkoryumin@fedora report]$ git add ЛОН2_Курилко-Рюмин Е.М_отчет
[emkurilkoryumin@fedora report]$ git commit -m "Add existing file"
[master 4722efc] Add existing file
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 labs/lab02/report/E.М_отчет
create mode 100644 labs/lab02/report/ЛОН2_Курилко-Рюмин
[emkurilkoryumin@fedora report]$
```

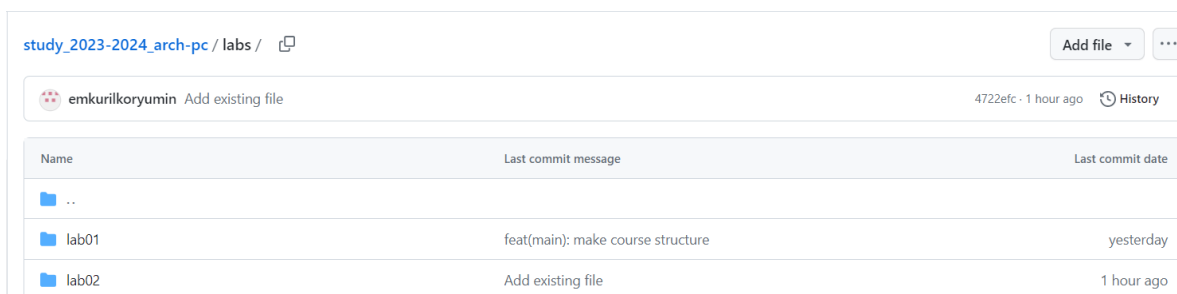
Рис. 4.19: Добавление файла на сервер

Отправляю в центральный репозиторий сохраненные изменения командой `git push -f origin master` (рис.21)

```
[emkurilkoryumin@fedora report]$ git push -f origin master
Перечисление объектов: 67, готово.
Подсчет объектов: 100% (67/67), готово.
Сжатие объектов: 100% (59/59), готово.
Запись объектов: 100% (67/67), 359.44 КиБ | 2.88 МиБ/с, готово.
Всего 67 (изменений 8), повторно использовано 26 (изменений 1), повторно использовано пакетов 0
remote: Resolving deltas: 100% (8/8), done.
To github.com:emkurilkoryumin/study_2023-2024_arch-pc.git
+ e0d8f07...4722efc master -> master (forced update)
[emkurilkoryumin@fedora report]$
```

Рис. 4.20: Отправка в центральный репозиторий сохраненных изменений

Проверяю на сайте GitHub правильность выполнения заданий. Вижу, что пояснение к совершенным действиям отображается (рис. 22)



study_2023-2024_arch-pc / labs /			Add file	...
emkurilkoryumin Add existing file			4722efc · 1 hour ago	History
Name	Last commit message	Last commit date		
..				
lab01	feat(main): make course structure	yesterday		
lab02	Add existing file	1 hour ago		

Рис. 4.21: Отправка в центральный репозиторий сохраненных изменений

При просмотре изменений так же вижу, что были добавлены файлы с отчетами по лабораторным работам (рис. 23)

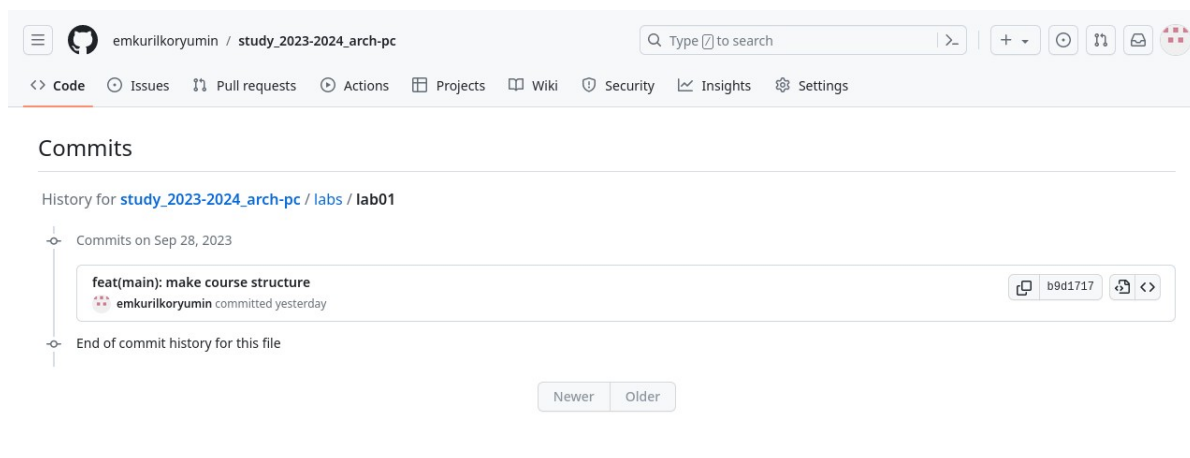


Рис. 4.22: Страница последних изменений в репозитории

Вижу, что отчеты по лабораторным работам находятся в соответствующих каталогах репозитория: отчет по первой - в lab01/report, по второй – в lab02/report (рис. 24)

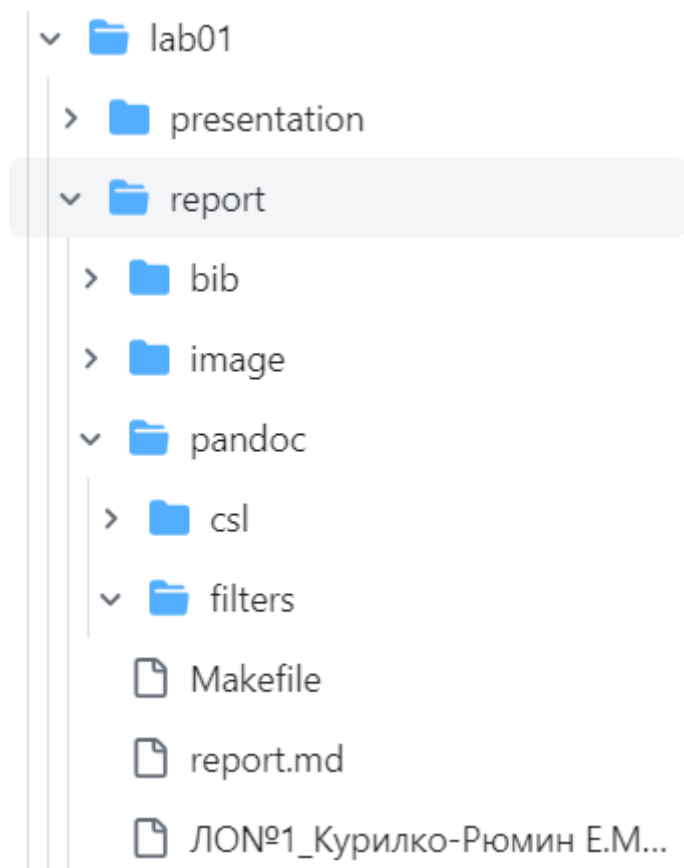


Рис. 4.23: Проверка каталогов

5 Выводы

Я приобрел практический опыт работы с системой git, изучил принципы и применение контроля версий.

Список литературы

1. Архитектура ЭВМ (rudn.ru)
2. Инструкция по использованию Git