Отчёта по лабораторной работе №2

Система контроля версии Git

Чекмарев Александр Дмитриевич | группа: НПИбд 02-23

Содержание

# 1 Цель работы

Целью работы является изучить идеологию и применение средств контроля версий. Приобрести практические навыки по работе с системой git (синхронизация github c Linux).

# 2 Выполнение лабораторной работы (Ход работы)

**2.1 Настройка github**

Создайте учётную запись на сайте https://github.com/ и заполните основные данные.

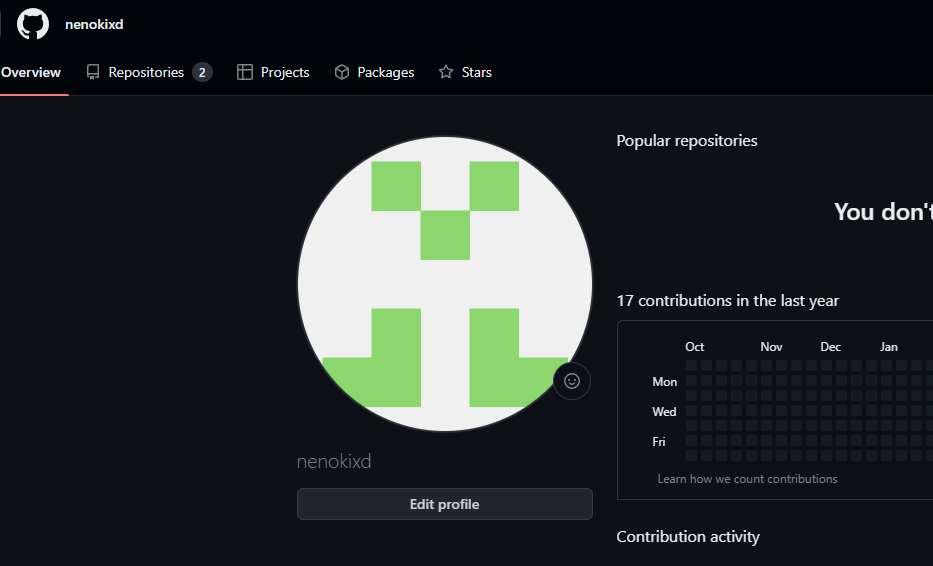


Рис 2.1.1: Демонстрация профиля на github

У меня уже был создан профиль на github.com, так что мне нужно было просто войти в него

**2.2 Базовая настройка github**

Сначала сделаем предварительную конфигурацию git. Откроем терминал и введём следующие команды, указав имя и email своего репозитория

Рис 2.2.1: Демонстрация ввода имени пользователя с email’ом

Рис 2.2.1: Демонстрация ввода имени пользователя с email’ом

Настроим utf-8 в выводе сообщений git

Рис 2.2.2: Настройка utf-8 с помощью команды

Рис 2.2.2: Настройка utf-8 с помощью команды

Зададим имя начальной ветки (будем называть её master)

Рис 2.2.3: Создание ветки под названием master

Рис 2.2.3: Создание ветки под названием master

Параметр autocrlf и safecrlf

Рис 2.2.4: Демонстрация ввода параметров в терминале

Рис 2.2.4: Демонстрация ввода параметров в терминале

**2.3 Создание SSH ключа**

Для последующей идентификации пользователя на сервере репозиториев необходимо сгенерировать пару ключей (приватный и открытый)

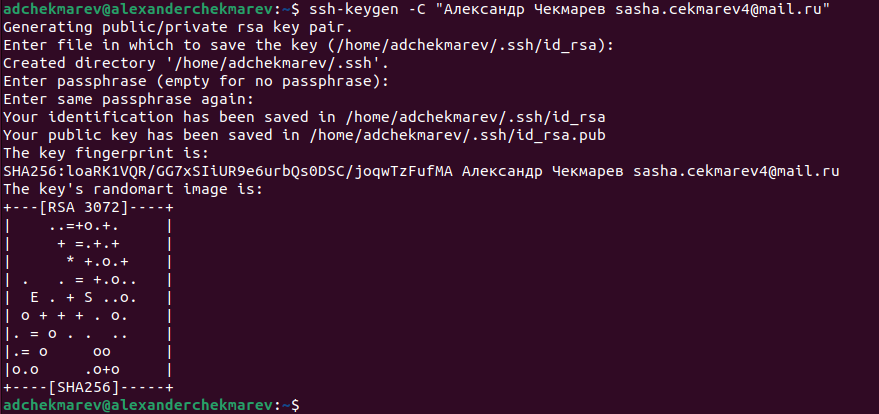


Рис 2.3.1: Демонстрация генерации ключей с помощью конмады shh-keygen -C

Ключи сохраняться в каталоге ~/.ssh/. Далее необходимо загрузить сгенерированный открытый ключ. Для этого зайти на сайт http://github.org/ под своей учётной записью и перейти в меню Setting. После этого выбрать в боковом меню SSH and GPG keys и нажать кнопку New SSH key. Скопировав из локальной консоли ключ в буфер обмена

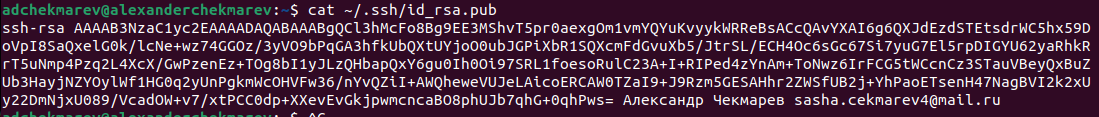


Рис 2.3.2: Демонстрация ключа в терминале

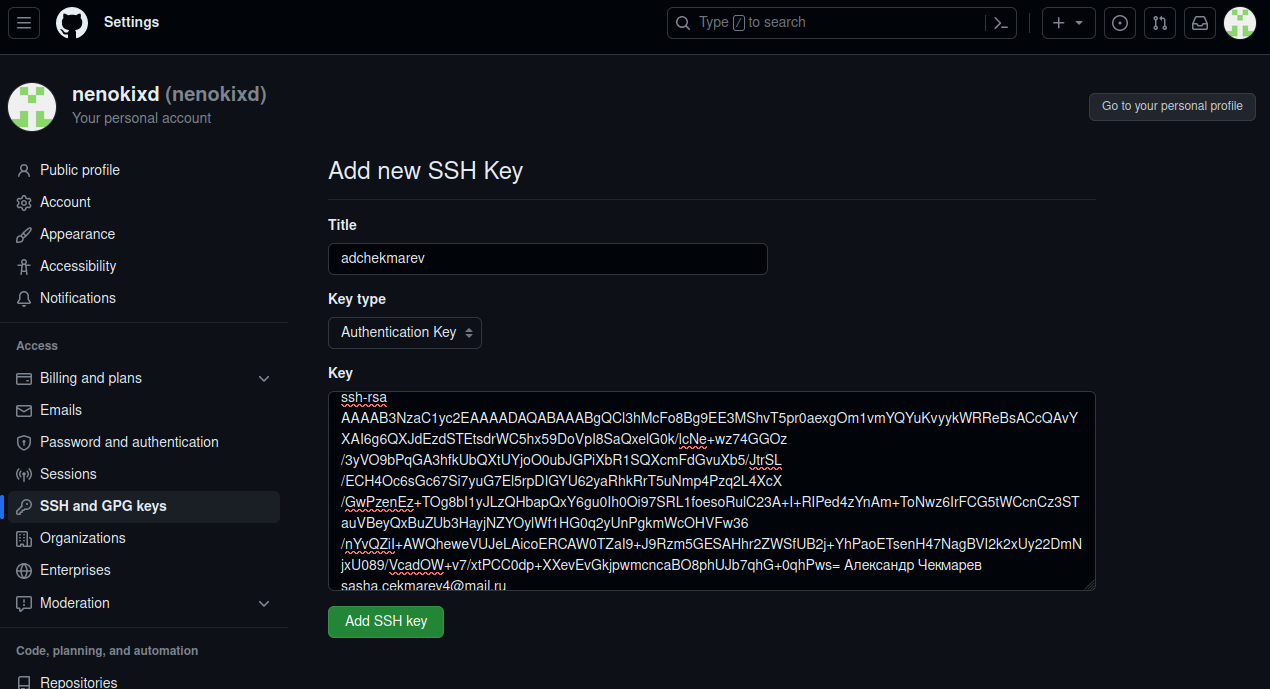


Рис 2.3.3: Копирование и вставление ключа с вводом имени в Title

**2.4 Сознание рабочего пространства и репозитория курса на основе шаблона**

При выполнении лабораторной работ следует придерживаться структуры рабочего пространства. Рабочее пространство по предмету располагается в следующей иерархии: ~/work/study/

\_\_\_ < учебный год>/

└── <название предмета>/  
  
 └── <код предмета>/

Например, для 2023–2024 учебного года и предмета «Архитектура компьютера» (код предмета arch-pc) структура каталогов примет следующий вид: ~/work/study/

\_\_\_ 2023–2024/

\_\_\_ Архитектура компьютера/

└── arch-pc/   
  
 └── labs/   
  
 └── lab01/   
  
 └── lab02/   
  
 └── lab03/  
  
 ...

• Каталог для лабораторных работ имеет вид labs.

• Каталоги для лабораторных работ имеют вид lab, например: lab01, lab02 и т.д. Название проекта на хостинге git имеет вид: study\_\_ Например, для 2023–2024 учебного года и предмета «Архитектура компьютера» (код предмета arch-pc) название проекта примет следующий вид: study\_2023–2024\_arch-pc

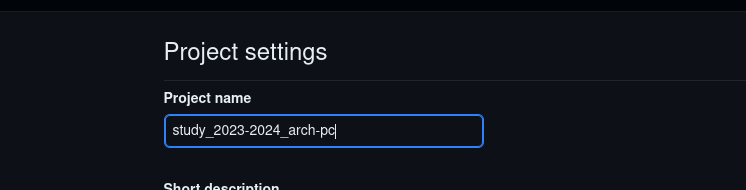


Рис 2.4.1: Демонстрация нужного названия проекта для хостинга git

Откроем терминал и создадим каталог для предмета «Архитектура компьютера»

Рис 2.4.2: Создание каталога с подкаталогами с помощью ключа -p

Рис 2.4.2: Создание каталога с подкаталогами с помощью ключа -p

**2.5 Сознание репозитория курса на основе шаблона**

Репозиторий на основе шаблона можно создать через web-интерфейс github. Перейдём на станицу репозитория с шаблоном курса https://github.com/yamadharma/course-directory-student-template. Далее выберем Use this template

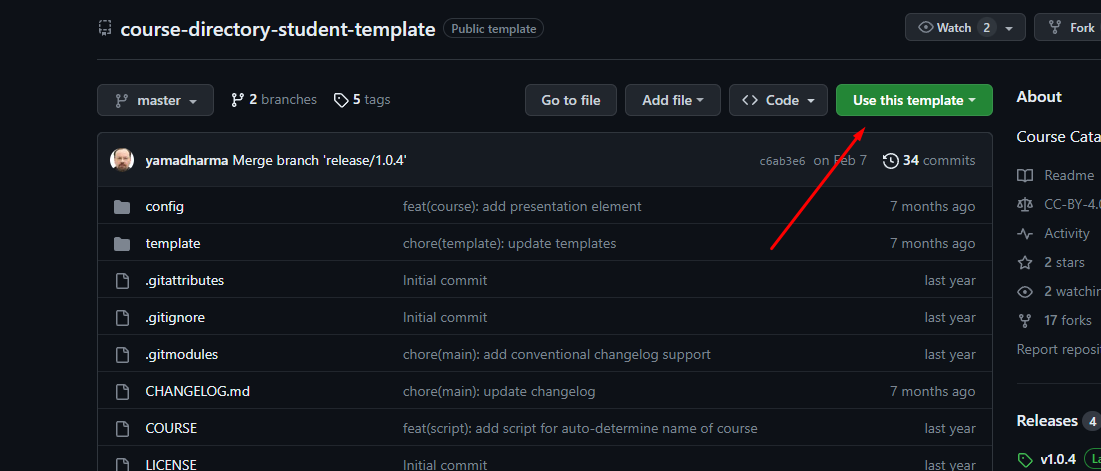


Рис 2.5.1: Демонстрация страницы с шаблоном курса и его использование для создания своего репозитория

В открывшемся окне зададим имя репозитория (Repository name) study\_2023–2024\_arh-pc и создадим репозиторий (кнопка Create repository from template)

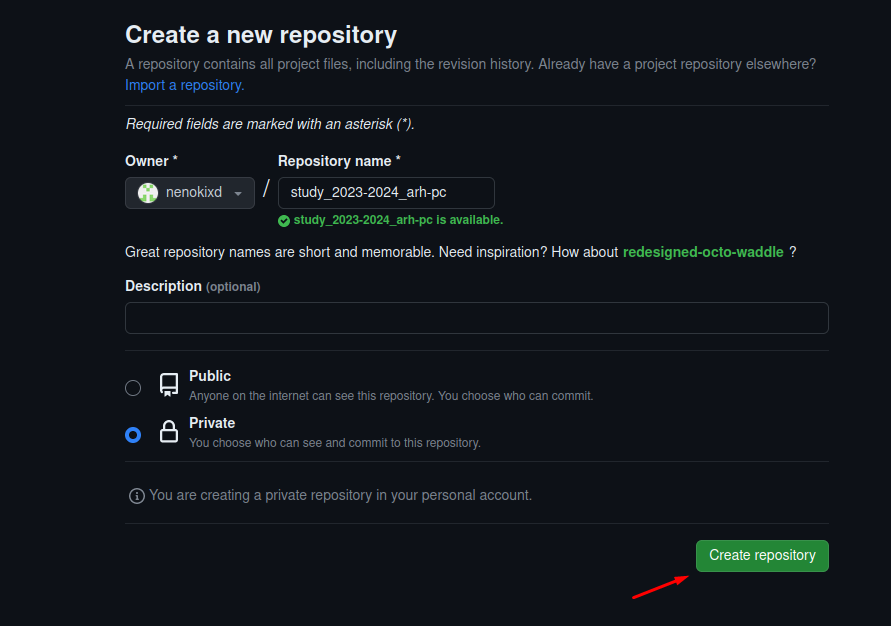
 Откроем терминал и перейдём в каталог курса:

Рис 2.5.3: Переход в каталог «Архитектура компьютера»

Рис 2.5.3: Переход в каталог «Архитектура компьютера»

Создадим каталог arсh-pc:

Рис 2.5.4: Создание каталога arсh-pc

Рис 2.5.4: Создание каталога arсh-pc

Ссылку для клонирования можно скопировать на странице созданного репозитория Code -> SSH:

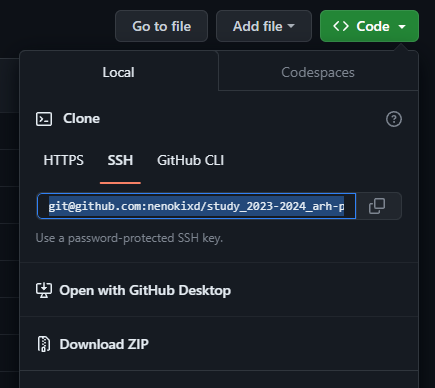


Рис 2.5.5: Демонстрация ссылки для копирования

Клонируем созданный репозиторий в arсh-pc:

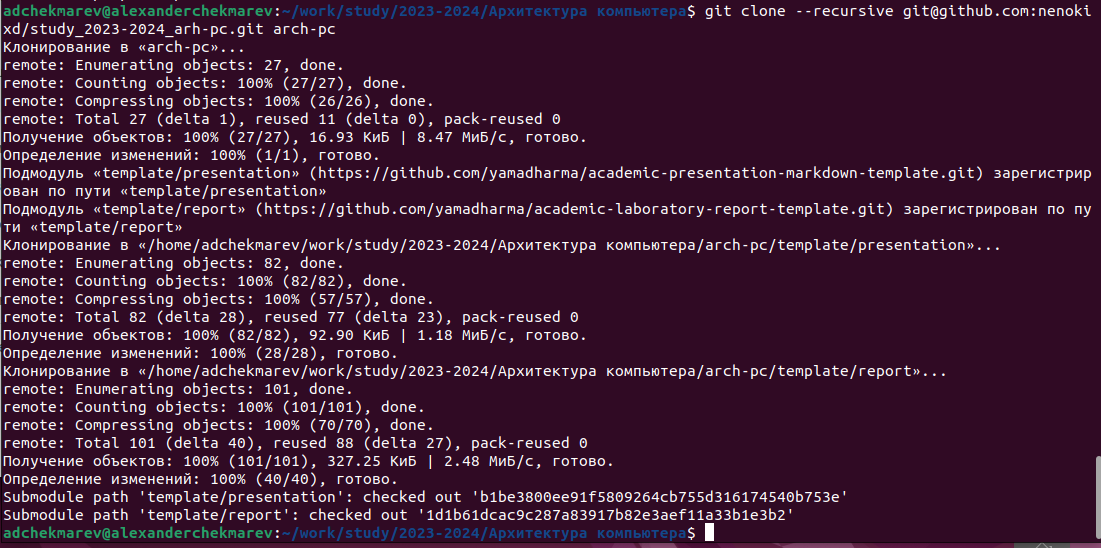


Рис 2.5.6: Клонирование репозитория

**2.6 Настройка каталога курса**

Перейдём в каталог курса:

Рис 2.6.1: Переход в каталог arсh-pc

Рис 2.6.1: Переход в каталог arсh-pc

Удалим лишние файлы:

Рис 2.6.2: Удаление лишнего файла package.json

Рис 2.6.2: Удаление лишнего файла package.json

Создадим необходимые каталоги:

Рис 2.6.3: Создание необходимых каталогов

Рис 2.6.3: Создание необходимых каталогов

Отправим файлы на сервер:

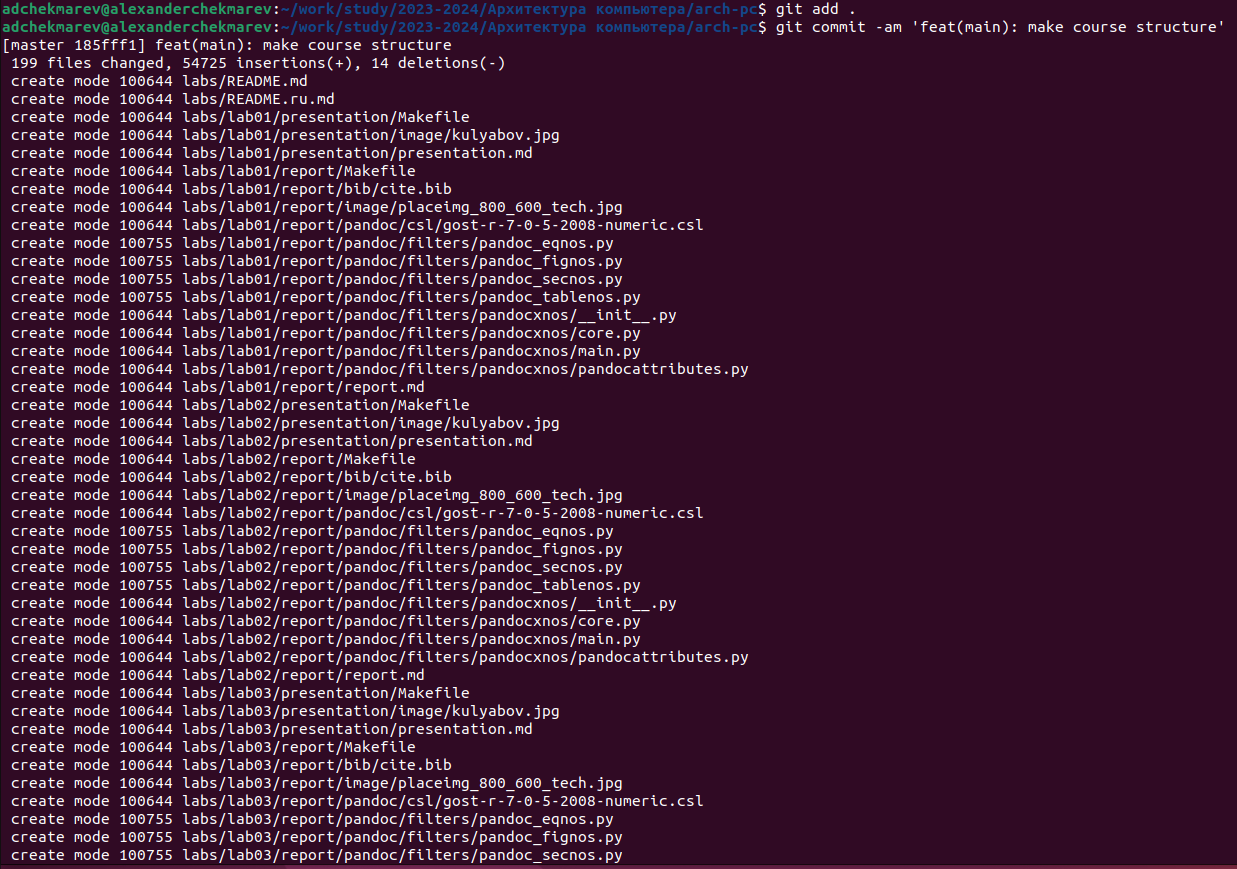


Рис 2.6.4: Отправка файлов на github

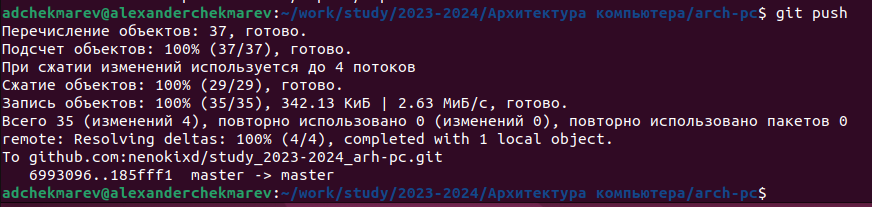


Рис 2.6.5: Отправка файлов на github

Проверим правильность создания иерархии рабочего пространства в локальном репозитории и на странице github:

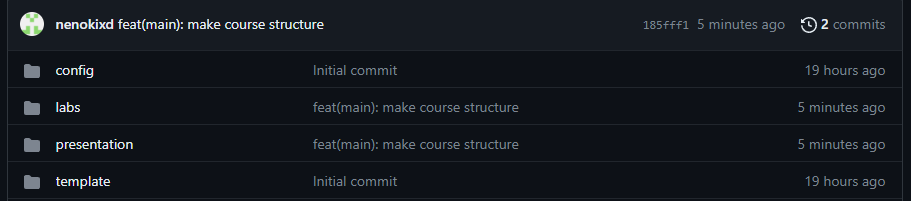


Рис 2.6.6: Проверка файлов на странице github

# 3 Самостоятельная работа

**Задание№1 Создайте отчет по выполнению лабораторной работы в соответствующем каталоге рабочего пространства (labs > lab02 > report).**

Создадим отчет по выполнению лабораторной работы в Домашней папке:

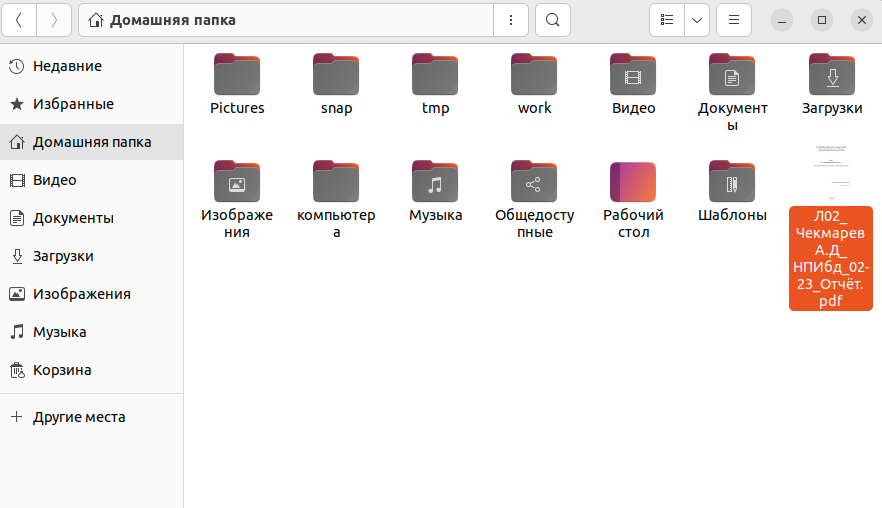


Рис 3.1.1: Демонстрация созданного отчета лабораторной работы в формате pdf

Копируем нужный нам файл из Домашней папки в каталог рабочего пространства /lab02/report:

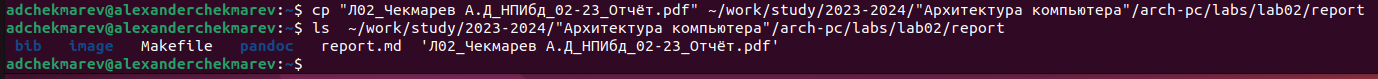


Рис 3.1.2: Копирование отчета из Домашней папки в нужный каталог и проверка выполненных действий

**Задание№2 Скопируйте отчеты по выполнению предыдущих лабораторных работ в соответствующие каталоги созданного рабочего пространства**

Скопируем отчет по первой лабораторной работе в каталог /lab01/report:

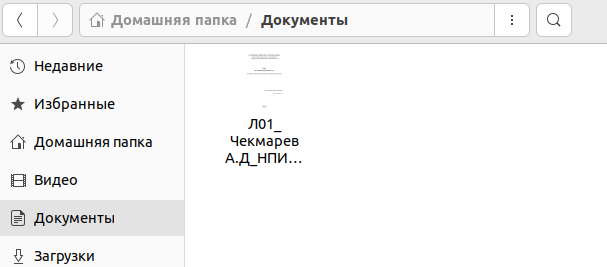


Рис 3.2.1: Демонстрация изначального каталога у отчета первой лаб. работы

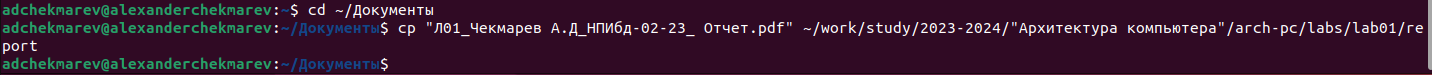


Рис 3.2.2: Копирование первой лаб. работы в нужный каталог

Проверим правильность выполненных действий:

Рис 3.2.3: Проверка выполненных действий

Рис 3.2.3: Проверка выполненных действий

**Задание№3 Загрузите файлы на github.**

Используем известные мне команды в терминале для загрузки файлов на github:

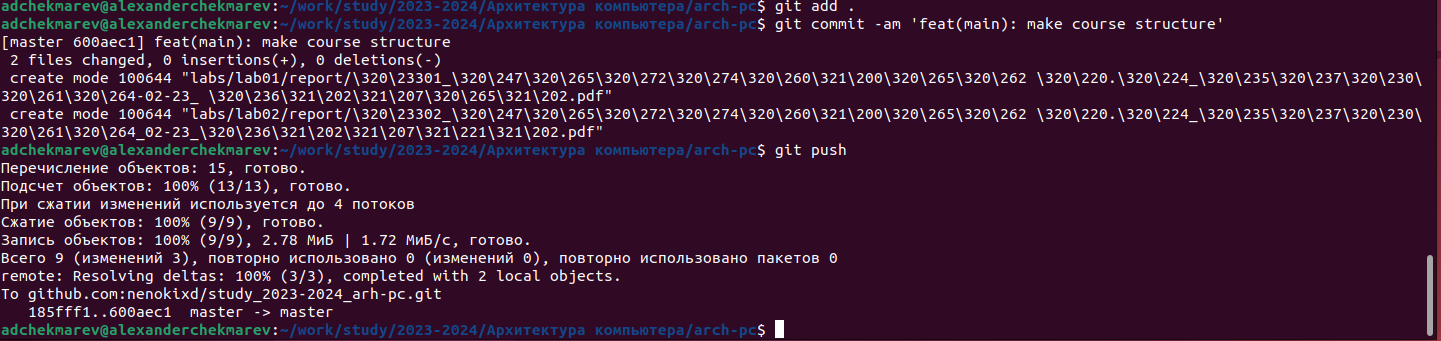


Рис 3.3.1: Демонстрация загрузки файлов на github

Проверим правильность выполненных действий:



Рис 3.3.2: Проверка загрузки файлов

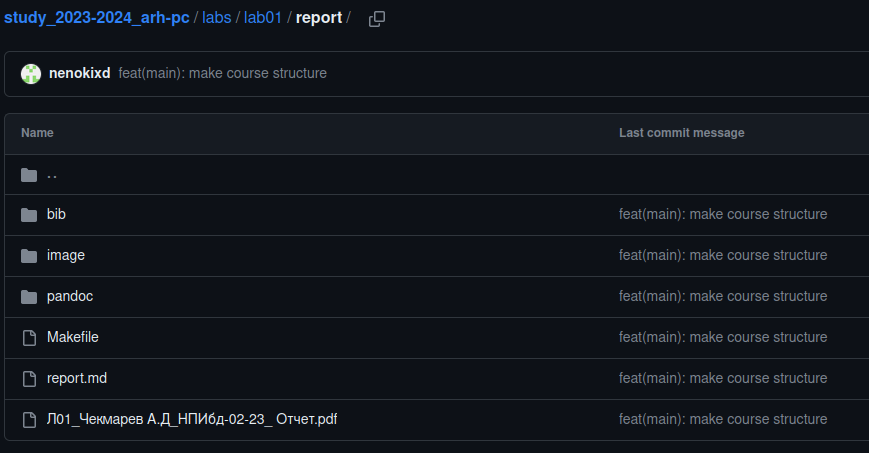


Рис 3.3.3: Проверка загрузки файлов

# 4 Вывод

В ходе выполнения лабораторной работы были изучены идеологии и применение средств контроля версий. Также были приобретены практические навыки по работе с системой git.