Лабораторная работа №2

Система контроля версий Git

Акунаева Антонина Эрдниевна

Содержание

# 1 1 Цель работы

Изучить идеологию и применение средств контроля версий. Приобрести практические навыки по работе с системой git.

# 2 2 Задание

Базовая настройка github и git.  
Создание ключа SSH.  
Создание репозитория курса на основе шаблона.  
Настройка каталога курса.

# 3 3 Выполнение лабораторной работы

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

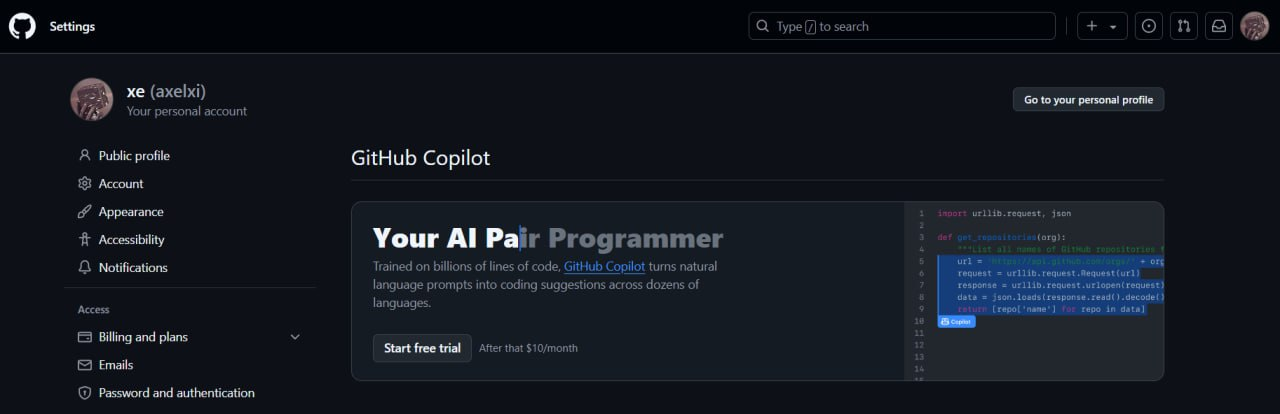


Рис. 1: Аккаунт Github

Аккаунт зарегистрирован на официальном сайте GitHub, почта подтверждена.

Основные данные для работы заполнены.

1. Откройте терминал и введите следующие команды, указав имя и email владельца репозитория:

git config –global user.name “”

git config –global user.email “work@mail”.

Задаём имя репозитория

Рис. 2: Задаём имя репозитория

Задаём email репозитория

Рис. 3: Задаём email репозитория

Прописываем указанные команды для настройки имени пользователя и почты аккаунта с помощью config от git с параметром –global для записи на глобальном уровне.

1. Настроим utf-8 в выводе сообщений git.

Настройка вывода сообщений git

Рис. 4: Настройка вывода сообщений git

Для корректного вывода информации (например, написанных на кириллице имён файлов) глобально переключим параметр quotepath секции core на false.

1. Зададим имя начальной ветки (будем называть её master).

Задаём имя начальной ветки

Рис. 5: Задаём имя начальной ветки

С помощью предписанной команды init.defaultBranch по её определению создаём основную ветку master.

1. Параметр autocrlf. Параметр safecrlf.

Настраиваем параметры git

Рис. 6: Настраиваем параметры git

Настраиваем параметры git

Рис. 7: Настраиваем параметры git

Значение заданного параметра autocrlf изменяем на input, что позволит заменять символы конца строки CRLF на LF при записи файла в репозиторий git и ничего не будет делать при чтении файла из репозитория.

Значение параметра safecrlf изменяем затем на warn, что позволит получить предупреждение о замене CRLF (с последующим переходом).

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

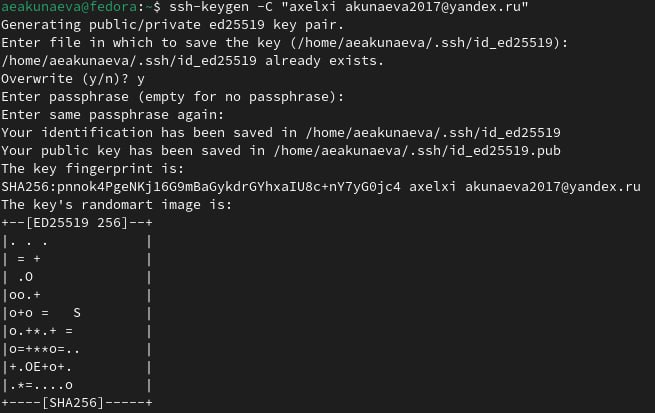


Рис. 8: Генерируем ключ SSH

Сгенерируем SSH ключи при помощи утилиты shh-keygen, обозначив имя и почту пользователя, а также ключ -C для запроса на изменение комментария, что позволит получить более развёрнутую инструкцию по местонахождению ключей.

1. Скопируйте из локальной консоли ключ в буфер обмена.

Использование команды cat

Рис. 9: Использование команды cat

Скопируем в буфер обмена ключ, использовав команду для объединения данных файлов — cat с указанием пути до файла с ключом из комментария в предыдущем задании, также обозначив команду для копирования в буфер обмена xclip -sel clip.

1. Далее необходимо загрузить сгенерённый открытый ключ, скопировав из локальной консоли ключ в буфер обмена. 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]. Вставляем ключ в появившееся на сайте поле и указываем для ключа имя (Title).

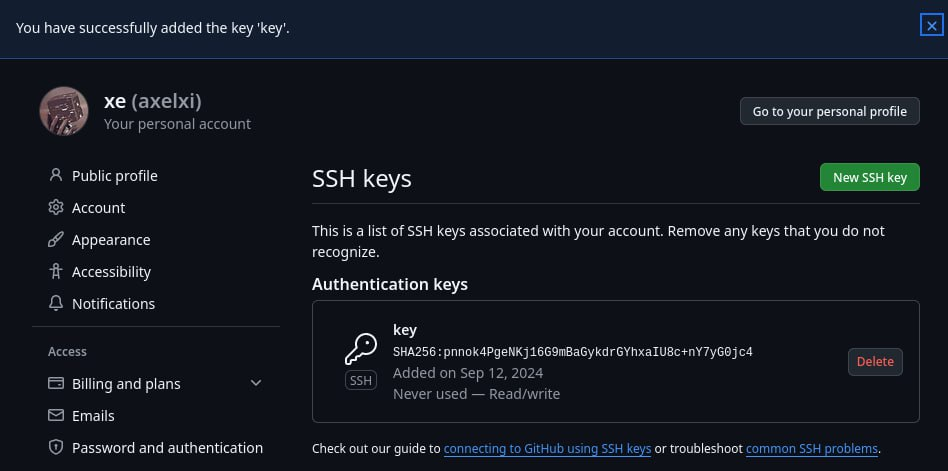


Рис. 10: SSH ключ на Github

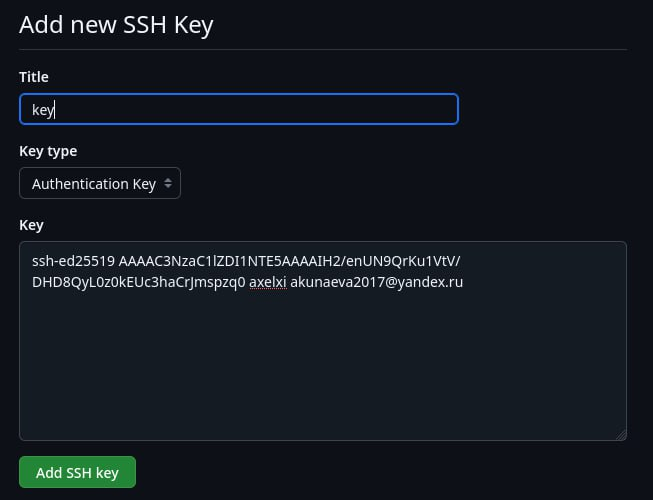


Рис. 11: Добавление SSH ключа на Github

Следуя инструкции, переходим на страницу для создания нового SSH ключа на GitHub, задаём название key для удобства и вставим ключ из буфера обмена.

1. Откройте терминал и создайте каталог для предмета «Архитектура компьютера».

Создание каталога с использованием mkdir

Рис. 12: Создание каталога с использованием mkdir

Создадим каталог для хранения файлов лабораторных и других заданий по предмету, использовав команду mkdir в терминале с ключом -p, чтобы создать вложенные каталоги, и пропишем нужный путь.

1. Перейдите на страницу репозитория с шаблоном курса https://github.com/yamadharma/course-directory-student-template. Далее выберите [Use this template]. В открывшемся окне задайте имя репозитория (Repository name) study\_2023–2024\_arhpc и создайте репозиторий (кнопка Create repository from template).

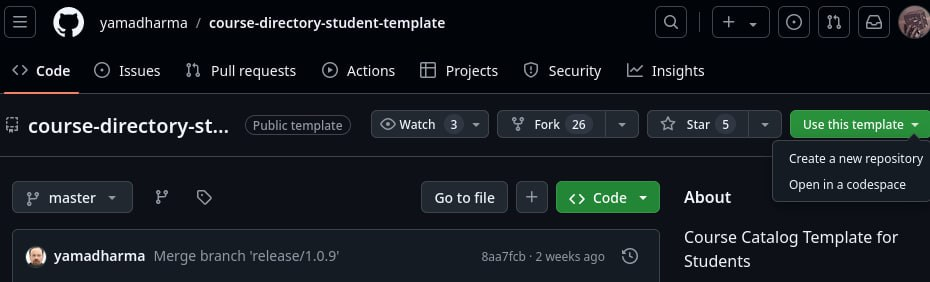


Рис. 13: Использование шаблона репозитория

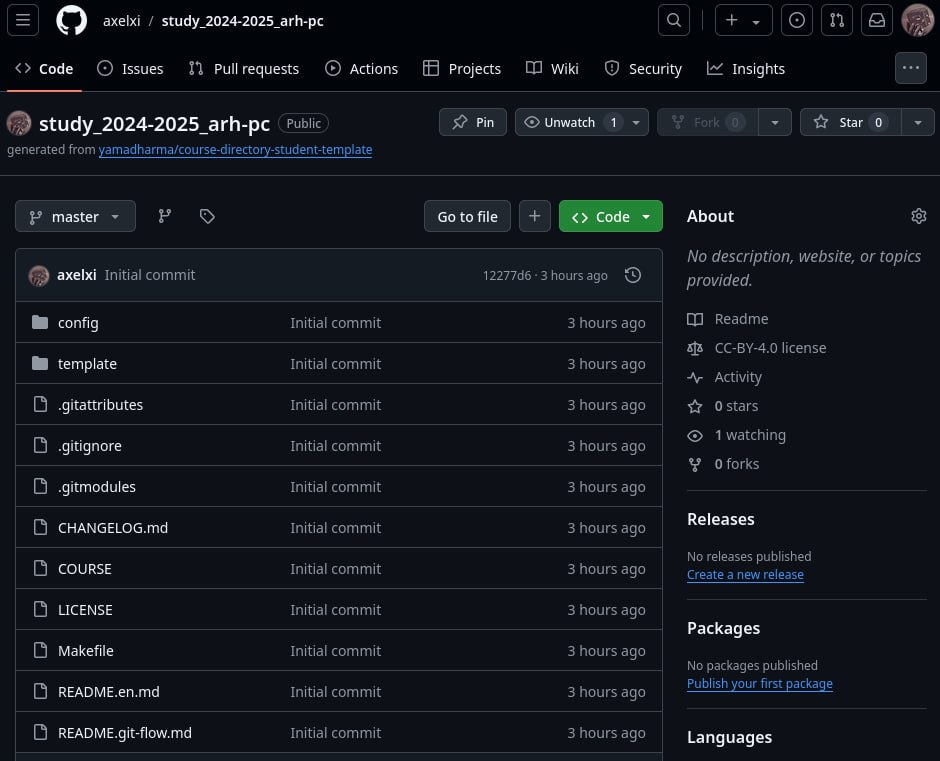


Рис. 14: Собственный шаблон репозитория

Переходим по ссылке к открытому репозиторию другого пользователя и используем как шаблон для создания своей копии репозитория, прописав требуемое имя для него. Репозиторий создан.

1. Откройте терминал и перейдите в каталог курса.

Использование команды cd

Рис. 15: Использование команды cd

Перейдём в каталог курса, указав его путь, при помощи команды cd.

1. Клонируйте созданный репозиторий.

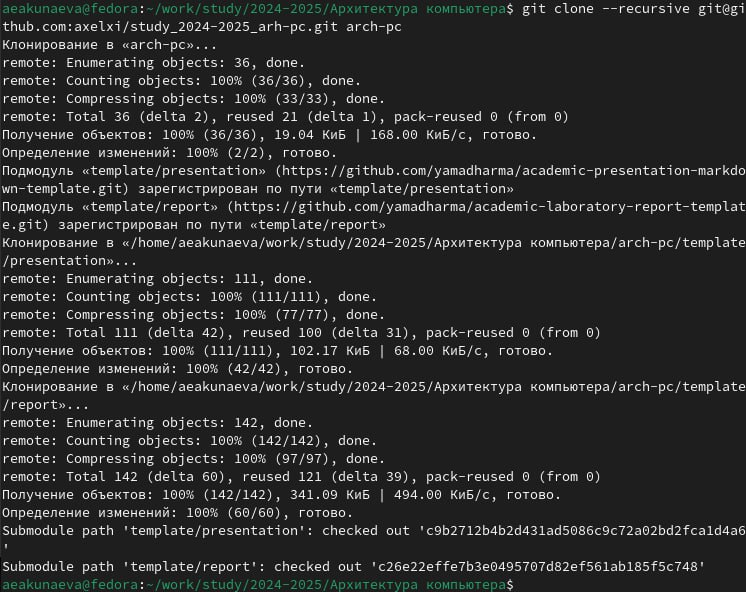


Рис. 16: Клонирование репозитория git

Клонируем репозиторий при помощи команды clone и ключом –recursive для рекурсивного клонирования (с подмодулями), после указав адрес репозитория на аккаунте и место назначения.

1. Перейдите в каталог курса.

Использование команды cd

Рис. 17: Использование команды cd

Снова перейдём в каталог курса при помощи cd.

1. Удалите лишние файлы.

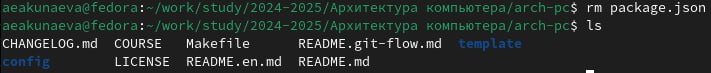
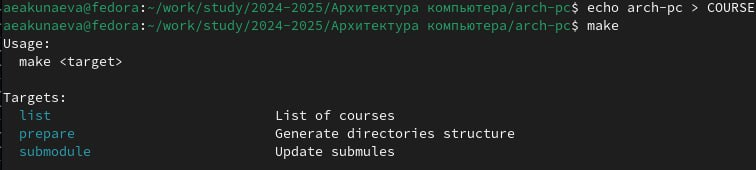


Рис. 18: Использование команд rm и ls

Удалим лишний файл package.json при помощи команды rm, пока мы находимся в каталоге курса. Проверим при помощи ls.

1. Создайте необходимые каталоги.

Создаём файл COURSE с записанной строкой arch-pc при помощи команды echo, использовав одиночный >. Для обозначенной команды make требуется действие, что указывается в комментарии, однако файл итак был создан (что подтверждено командой ls).

1. Отправьте файлы на сервер.

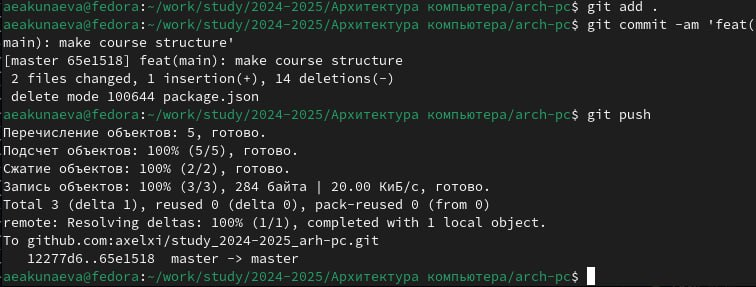


Рис. 19: Отправление изменений локального репозитория

Добавим все созданные каталоги и файлы при помощи git add ., после чего сохраним изменения следующей командой git commit -am с обозначением коммита в кавычках и отправим данные в центральный репозиторий из локального командой git push.

1. Проверьте правильность создания иерархии рабочего пространства в локальном репозитории и на странице github.

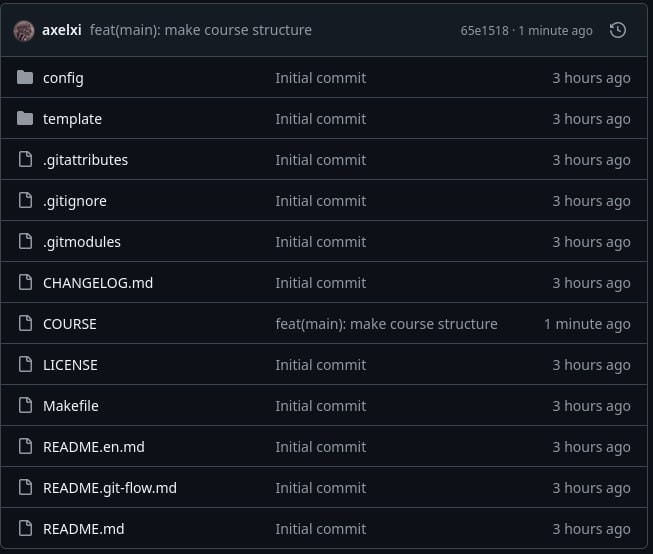


Рис. 20: Отправление изменений локального репозитория

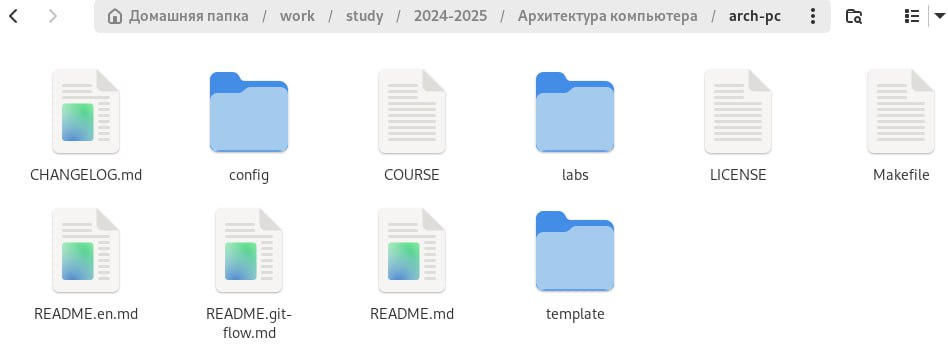


Рис. 21: Каталог курса

В локальном репозитории и на сайте GitHub пользователя присутствуют все указанные файлы в верно расставленной иерархии (папка labs из предыдущей лабораторной).

# 4 4 Описание результатов выполнения заданий для самостоятельной работы

1. Создайте отчет по выполнению лабораторной работы в соответствующем каталоге рабочего пространства (labs>lab02>report).

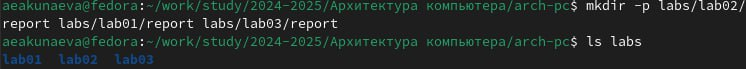


Рис. 22: Использование команд mkdir и ls

Использование команды cd

Рис. 23: Использование команды cd

Создаём каталоги для хранения лабораторных через терминал, используя mkdir и ключ -p, также проверим успешность выполнения команды при помощи ls.

1. Скопируйте отчеты по выполнению предыдущих лабораторных работ в соответствующие каталоги созданного рабочего пространства.



Рис. 24: Каталог labs/lab01/report

Перемещаем документ с отчётом в каталог labs/lab01/report.

1. Загрузите файлы на github.

Использование команды cd

Рис. 25: Использование команды cd

Переходим в каталог курса при помощи cd.

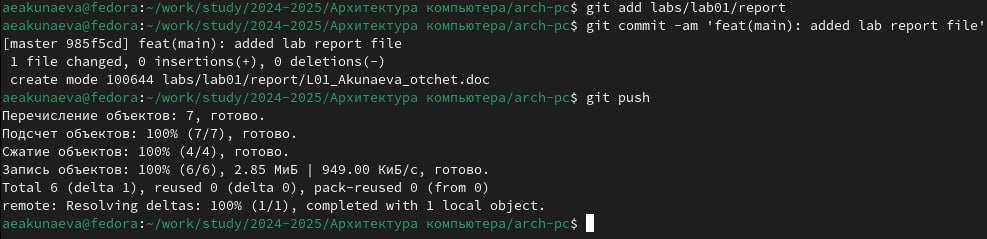


Рис. 26: Отправление изменений локального репозитория

Загружаем файлы на GitHub сначала при помощи команды git add, указав путь до каталога с отчётом к предыдущей лабораторной, затем сохраним при помощи commit -am, указав коммит. Отправляем данные в центральный репозиторий на сайт с git push.

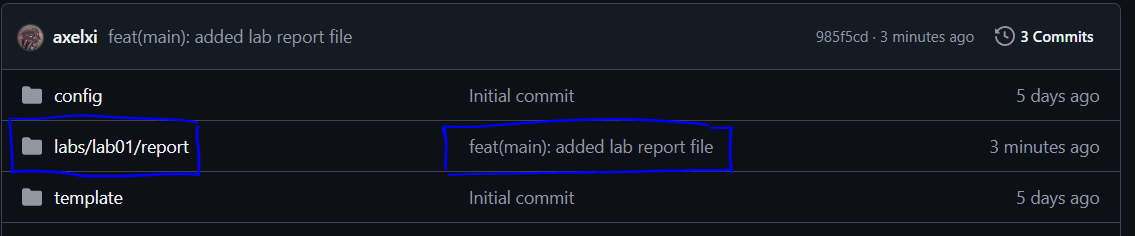


Рис. 27: Github репозиторий

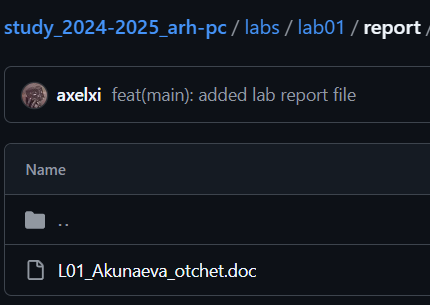


Рис. 28: Всё правильно

Проверим наличие каталогов с отчётом на сайте GitHub в репозитории лабораторной. Файл отчёта действительно располагается в подкаталоге report.

# 5 5 Выводы

Я изучила идеологию и применение средств контроля версий и приобрела практические навыки по работе с системой git.