Отчёт по лабораторной работе №2

Дисциплина: Архитектура компьютера

Кижваткина Анна Юрьевна

Содержание

# 1 Цель работы

Целью данной лабораторной работы является получение практических навыков работы с системой Git при помощи командной строки. В ходе неё мы я изучить идеологию и применение средств контроля версий.

# 2 Задание

1. Настройка github.  
2. Базовая настройка git.  
3. Создание SSH ключа.  
4. Создание рабочего пространства и репозитория курса на основе шаблона.  
5. Создание репозитория курса на основе шаблона.  
6. Настройка каталога курса.  
7. Выполнение заданий для самостоятельной работы.

# 3 Теоретическое введение

Здесь описываются теоретические аспекты, связанные с выполнением работы.

Например, в табл. 1 приведено краткое описание стандартных каталогов Unix.

Таблица 1: Описание некоторых каталогов файловой системы GNU Linux

| Имя каталога | Описание каталога |
| --- | --- |
| / | Корневая директория, содержащая всю файловую |
| /bin | Основные системные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям |
| /etc | Общесистемные конфигурационные файлы и файлы конфигурации установленных программ |
| /home | Содержит домашние директории пользователей, которые, в свою очередь, содержат персональные настройки и данные пользователя |
| /media | Точки монтирования для сменных носителей |
| /root | Домашняя директория пользователя root |
| /tmp | Временные файлы |
| /usr | Вторичная иерархия для данных пользователя |

Более подробно про Unix см. в [1–4].

# 4 Выполнение лабораторной работы

3.1 Настройка Github Создаю учетную запись на сайте Github. Заполняю основные данные для учетной записи.

3.2 Базовая настройка git. Открываю терминал, делаю предварительную конфигурацию git. Вводим команду git config –global user.name “” и указываем имя, следом вводим git config –global user.email ““, вписывая электронную почту аккаунта github. (рис. 1 и рис. 2)

Рис. 1: Команда git config –global user.name “”

Рис. 1: Команда git config –global user.name “”

Рис. 2: Команда git config –global user.email “”

Рис. 2: Команда git config –global user.email “”

Настраиваем utf-8 в выводе сообщений git. (рис. 3)

Рис. 3: Настройка utf-8

Рис. 3: Настройка utf-8

Задаем имя для начальной ветки. Она будет называться «master». (рис. 4)

Рис. 4: Создание начальной ветки

Рис. 4: Создание начальной ветки

Задаем параметры autocrlf и safecrlf для корректного выполнения команд. (рис. 5 и рис. 6)

Рис. 5: Параметр autocrlf

Рис. 5: Параметр autocrlf

Рис. 6: Параметр safecrlf

Рис. 6: Параметр safecrlf

3.3 Создание SSH ключа.

Для последующей идентификации пользователя на сервере репозиториев необходимо сгенерировать пару ключей (приватный и открытый). Для этого вводим в командную строку команду ssh-keygen –C “”, вводя имя пользователя и электронную почту. Ключ автоматически сохранится в ~/.ssh/. (рис. 7)

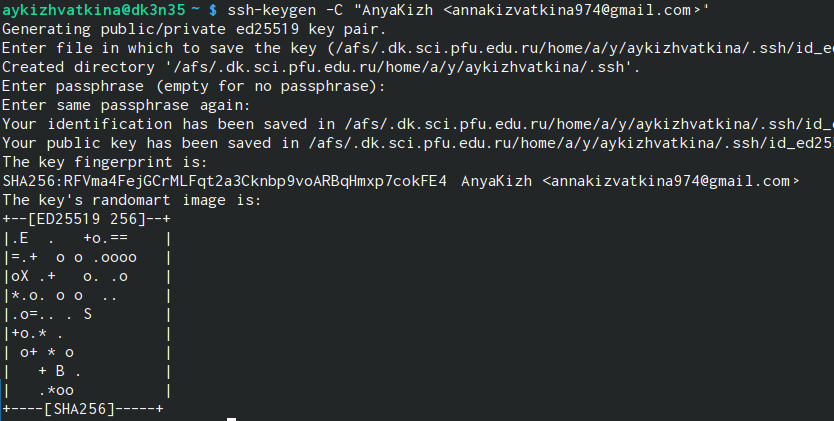


Рис. 7: Генерация ключа SSH

Далее загружаем сгенеренный открытый ключ. Заходим на сайт под своей учетной записью, переходим в меню, находим меню ключей и создаем новый ключ. (рис. 8)

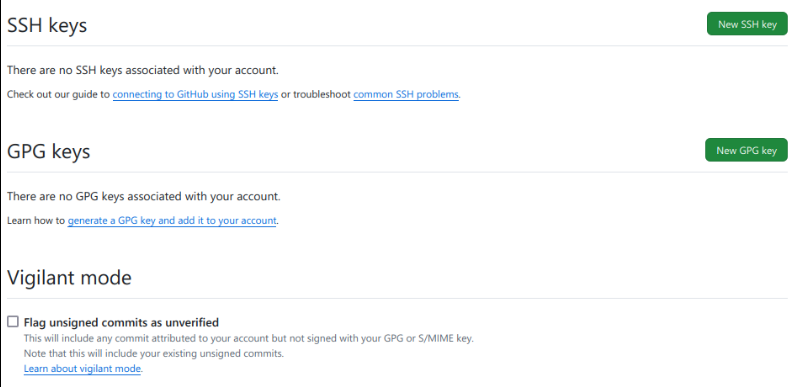


Рис. 8: Окно меню ключей

Копируем из локальной консоли ключ в буфер обмена. (рис. 9)

Рис. 9: Команда cat

Рис. 9: Команда cat

Вставляем ключ в поле сайта и указываем имя для ключа. (рис. 10)

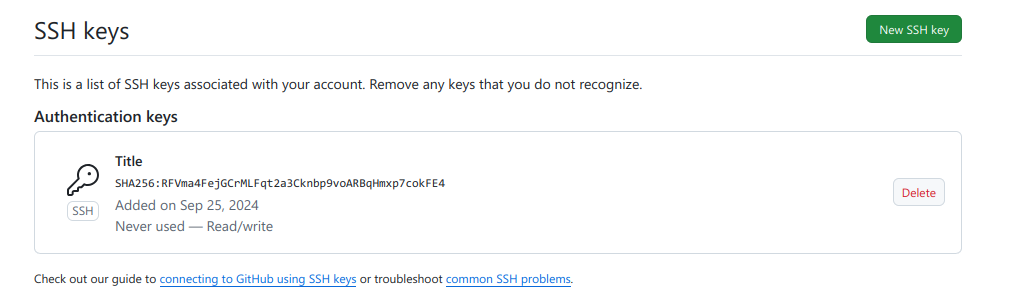


Рис. 10: Добавление ключа

3.4 Создание рабочего пространства и репозитория курса на основе шаблона.

Открываем терминал и создаем каталог для предмета “Архитектура компьютера”. (рис. 11)

Рис. 11: Создание каталога

Рис. 11: Создание каталога

3.5 Создание репозитория курса на основе шаблона.

Переходим на станицу github в репозиторий с шаблоном курса https://github.com/yamadharma/cour se-directory-student-template. Далее выбираем “Use this template”. (рис. 12)

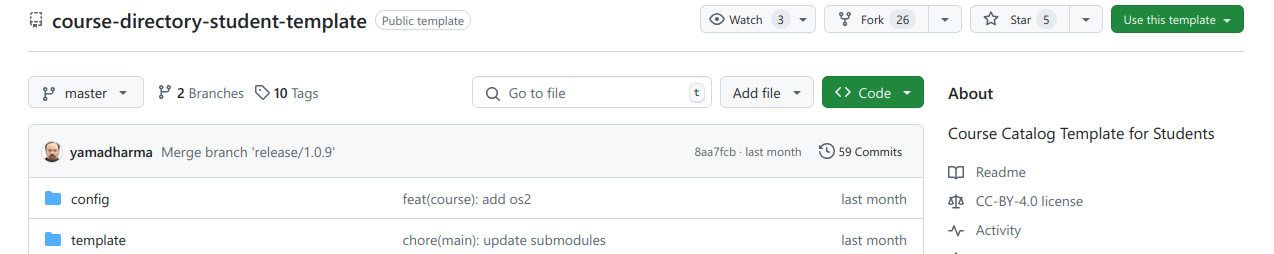


Рис. 12: Страница шаблона

В открывшемся окне задаем имя репозитория. Создаем репозиторий. Проверяем создался ли репозиторий. (рис. 13 и рис. 14)

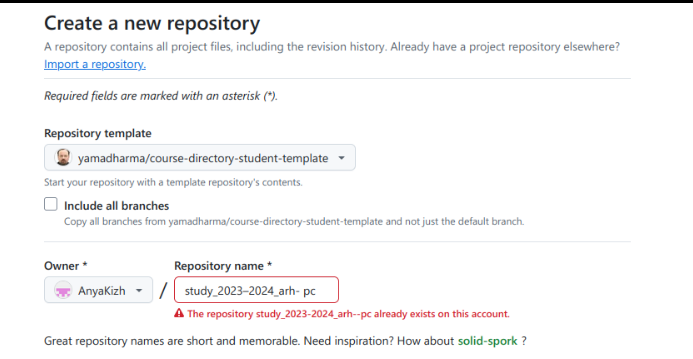


Рис. 13: Создание репозитория

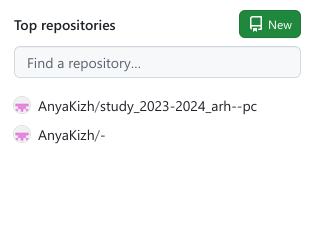


Рис. 14: Созданный репозиторий

Открываем терминал и переходим в каталог курса при помощи cd. (рис. 15)

Рис. 15: Перемещение в каталог

Рис. 15: Перемещение в каталог

Клонируем созданный репозиторий и помощи команды git clone –recursive ’’. (рис. 16)

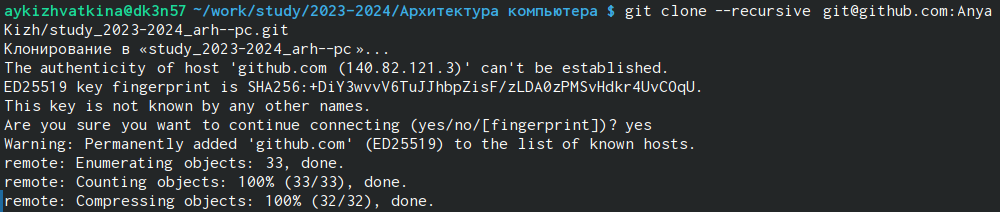


Рис. 16: Клонирование репозитория

Копируем ссылку для клонирования на странице созданного репозитория. Переходим в окно Code, следом в SSH. (рис. 17)

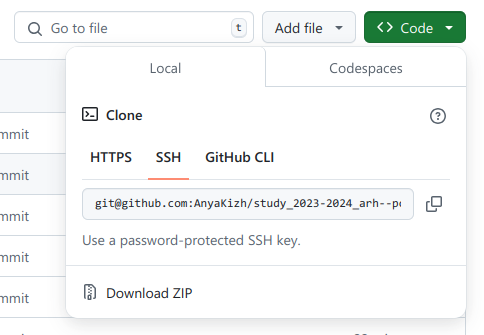


Рис. 17: Окно с ссылкой для копирования

3.6 Настройка каталога курса.

Переходим в каталог курса при помощи cd. (рис. 18)

Рис. 18: Перемещение между директориями

Рис. 18: Перемещение между директориями

Удаляем лишние файлы. (рис. 19)

Рис. 19: Удаление файлов

Рис. 19: Удаление файлов

Создаем необходимые каталоги. (рис. 20)

Рис. 20: Создание каталогов

Рис. 20: Создание каталогов

Отправляем файлы на сервер используя команды git add ., git commit –am и git push. (рис. 21 и рис. 22)

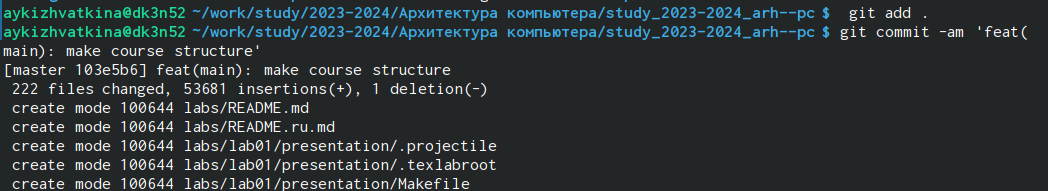


Рис. 21: Добавление и сохранение изменений

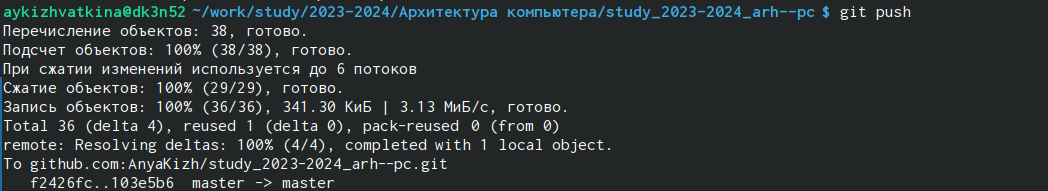


Рис. 22: Загрузка изменений на сайт

Проверяем правильность создания иерархии рабочего пространства в локальном репозитории и на странице github. (рис. 23)

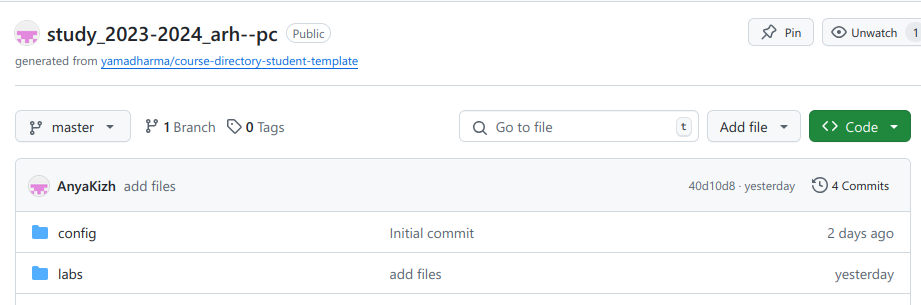


Рис. 23: Проверка репозитория

3.7 Выполнение заданий для самостоятельной работы.

Переходим в labs/lab02/report с помощью cd. (рис. 24)

Рис. 24: Перемещаемся в каталог labs/lab02/report

Рис. 24: Перемещаемся в каталог labs/lab02/report

Создаем файл для отчета по лабораторной работе, используя touch. (рис. 25)

Рис. 25: Создание файла

Рис. 25: Создание файла

Переходим из подкаталога lab02/report в подкаталог lab01/report. (рис. 26)

Рис. 26: Перемещаемся в каталог labs/lab02/report

Рис. 26: Перемещаемся в каталог labs/lab02/report

Копируем первую лабораторную с помощью cp и проверяем правильность выполнения команды. (рис. 27)

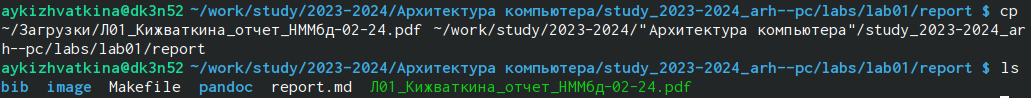


Рис. 27: Копируем первую лабораторную в каталог

Переходим в подкаталог lab02/report. Копируем вторую лабораторную в каталог и проверяем правильность выполнения. (рис. 28 и рис. 29)

Рис. 28: Перемещение в каталог

Рис. 28: Перемещение в каталог

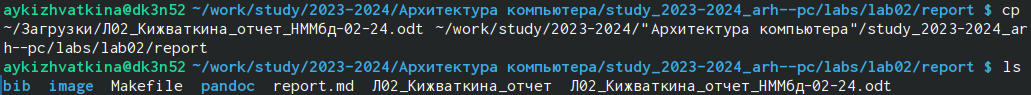


Рис. 29: Копируем вторую лабораторную в каталог

При помощи команды git add ’’ добавляем новые файлы. (рис. 30)

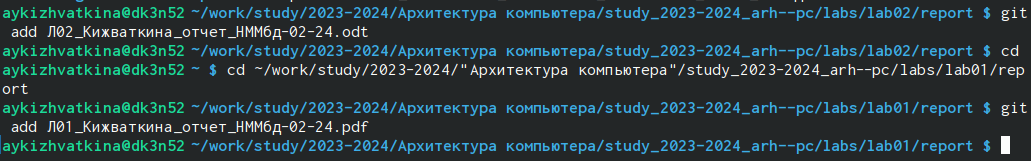


Рис. 30: Добавляем новые файлы

Сохраняем изменение при помощи команды git commit –am. Переносим в репозиторий сохраненные изменения командой git pish. (рис. 31)

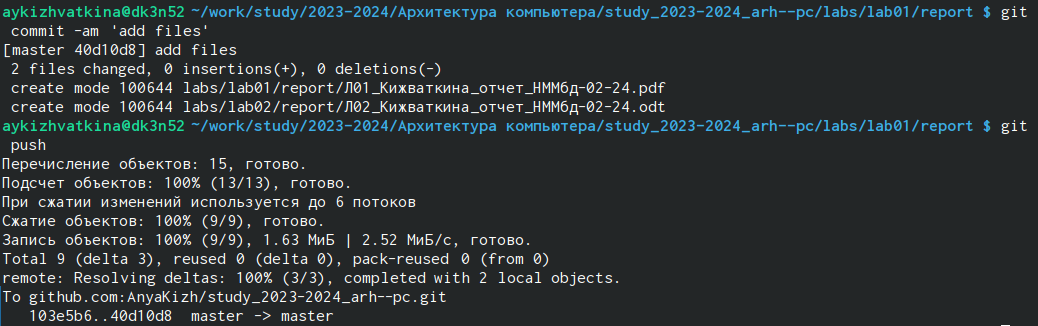


Рис. 31: Сохранение и перенос изменений

Проверяем на сайте правильность выполнения заданий. (рис. 32 и рис. 33)

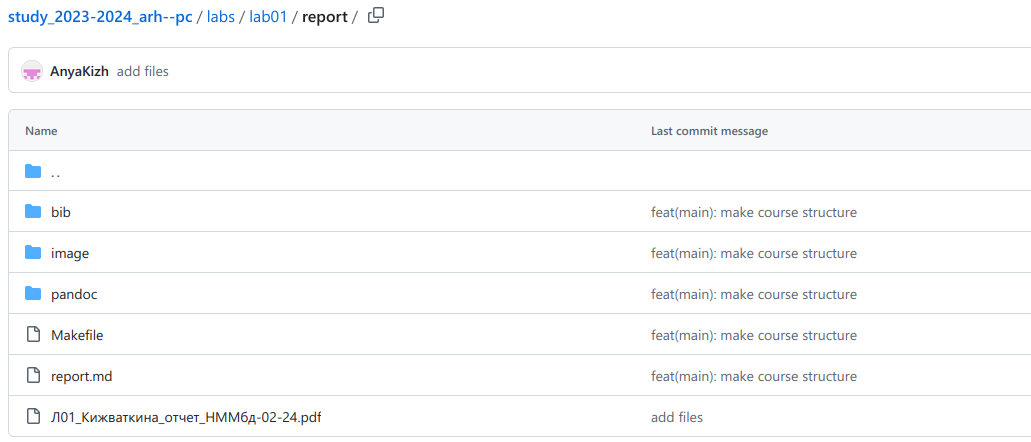


Рис. 32: Проверка labs/lab01/report

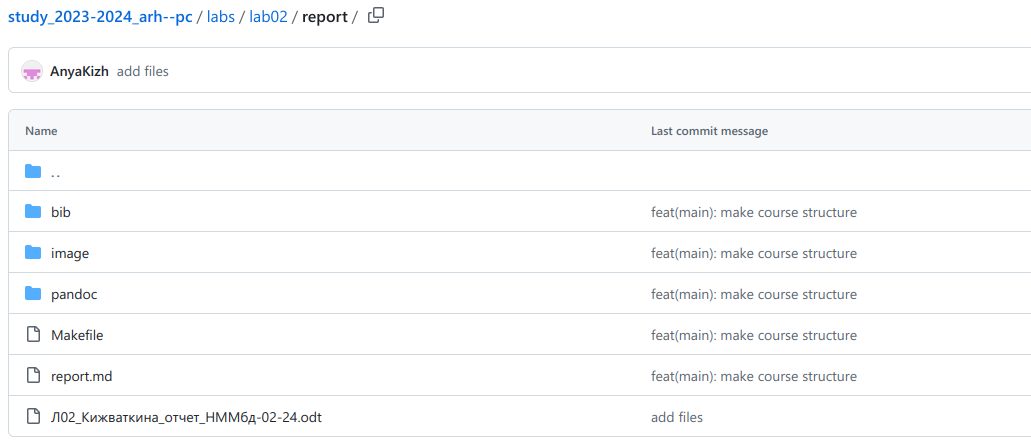


Рис. 33: Проверка labs/lab02/report

# 5 Выводы

При выполнении данной лабораторной работы мы изучили идеологию и применение средств контроля версий, а также приобрели практические навыки по работе с системой git.

# Список литературы

1. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. СПб.: Питер, 2015. 1120 с.

2. Robbins A. Bash Pocket Reference. O’Reilly Media, 2016. 156 с.

3. Zarrelli G. Mastering Bash. Packt Publishing, 2017. 502 с.

4. Newham C. [Learning the bash Shell: Unix Shell Programming](http://www.amazon.com/Learning-bash-Shell-Programming-Nutshell/dp/0596009658). O’Reilly Media, 2005. 354 с.