Отчёт по лабораторной работе 2

Дисциплина: Архитектура компьютера

Милдред Вакутайпа

Содержание

## 0.1 Цель работы

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

## 0.2 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.

## 0.3 Выполнение лабораторной работы

3.1. Настройка GitHub

Создаю учетную запись на сайте GitHub. Далее я заполнила основные данные учетной записи:

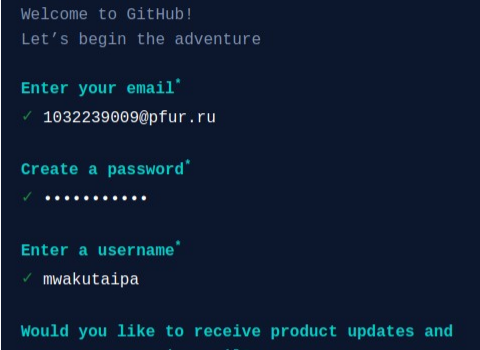


Рис 3.1.0

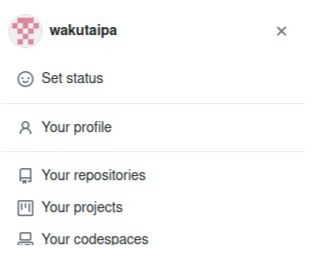


Рис 3.1.1

3.2. Базовая настройка Git

Открываю терминал и устанавливаю git с помощью sudo apt-get install git:

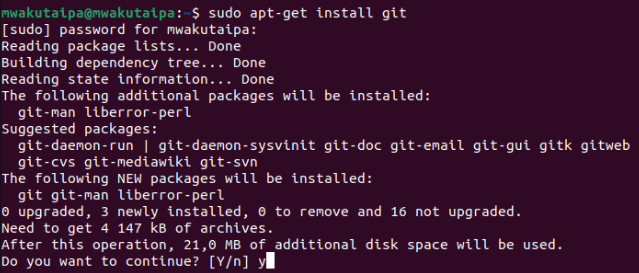


Рис 3.2.0

Потом 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 мою:

Рис 3.2.1

Рис 3.2.1

Настраиваю utf-8 в выводе сообщений git и Задаю имя «master» для начальной ветки:

Рис 3.2.2

Рис 3.2.2

Задаю параметр autocrlf со значением input и параметр safecrlf со значением warn:

Рис 3.2.3

Рис 3.2.3

3.3. Создание SSH-ключа

Для последующей идентификации пользователя на сервере репозиториев необходимо сгенерировать пару ключей (приватный и открытый). Для этого использую команду ssh-keygen -C “Имя , work@email”:

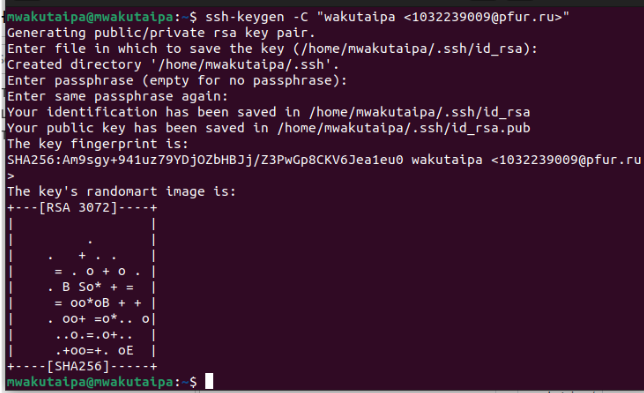


Рис 3.3.0

Xclip команд который помагает скопировать любой текст через терминал. Устанавливаю xclip с помощью команды sudo apt install xclip:

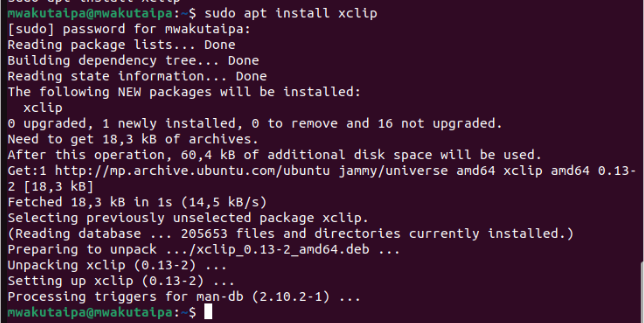


Рис 3.3.1

Копирую открытый ключ из директории, в которой он был сохранен, с помощью xclip:

Рис 3.3.2

Рис 3.3.2

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:

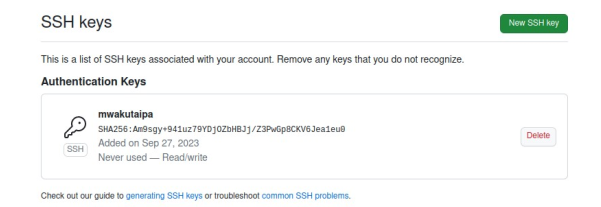


Рис 3.3.3

*3.4. Создание рабочего пространства и репозитория курса на основе шаблона*

Создаю директорию, рабочее пространство, с помощью mkdir. Проверяю с помощью ls, действительно ли были созданы каталоги:

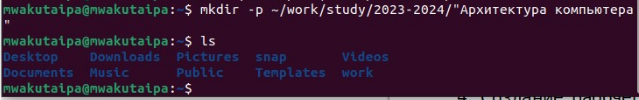


Рис 3.4.0

3.5. Создание репозитория курса на основе шаблона

Перехожу на станицу репозитория с шаблоном курса и выбираю «Use this template»:

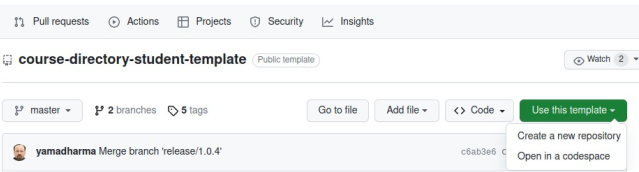


Рис 3.5.0

В открывшемся окне задаю имя репозитория (Repository name): study1\_2023–2024\_arhpc и создаю репозиторий, нажимаю на кнопку «Create repository from template»:

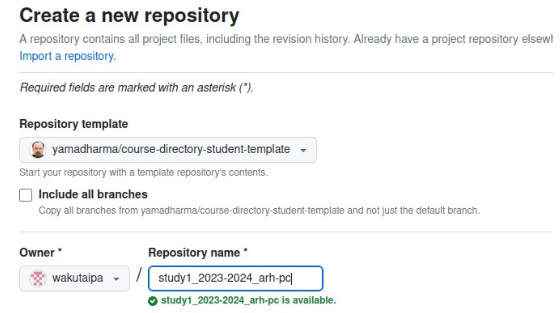


Рис 3.5.1

Через терминал перехожу в созданный каталог курса с помощью cd:

Рис 3.5.2

Рис 3.5.2

Клонирую созданный репозиторий с помощью команды git clone -- recursive git@github.com:/study\_2023–2024\_arh-pc.git arch-pc:

Рис 3.5.3

Рис 3.5.3

3.6. Настройка каталога курса

Перехожу в каталог arch-pc с помощью cd:

Рис 3.6.0

Рис 3.6.0

Удаляю package.json с помощью rm:

Рис 3.6.1

Рис 3.6.1

Создаю необходимые каталоги с помощью echo:

Рис 3.6.2

Рис 3.6.2

Отправляю созданные каталоги на сервер с git add ., git commit и git push:

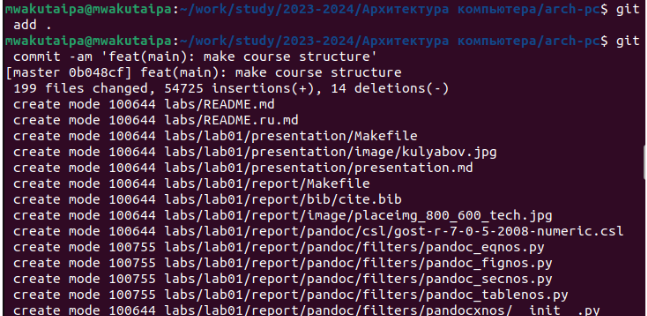


Рис 3.6.3

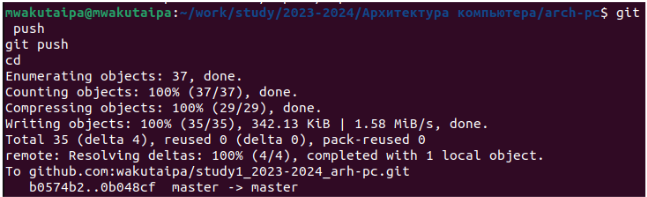


Рис 3.6.4

Проверяю правильность выполнения работы в локальном репозетории и на GitHub:

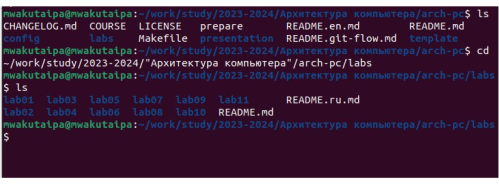


Рис 3.6.5

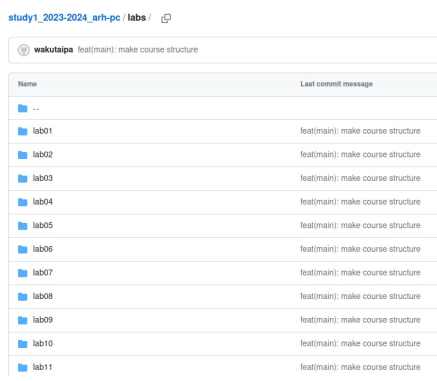


Рис 3.6.6

## 0.4 Выполнение заданий для самостоятельной работы

4.1. Перехожу в директорию labs/lab02/report с помощью cd. Создаю в каталоге файл для отчета по второй лабораторной работе с помощью touch:

Рис 4.1.0

Рис 4.1.0

Оформить отчет я смогу с помощью текстового редактора LibreOffice Writer. Захожу в файл, который создала и выбираю Open With Other Application, LibreOffice Writer, а потом создаю отчет lab2 (позже сохраню его как lab2.odt):

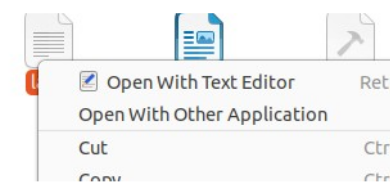


Рис 4.1.1

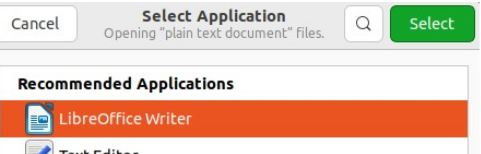


Рис 4.1.2

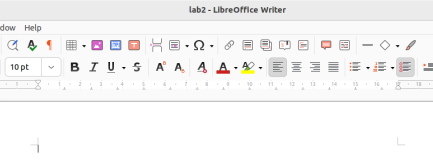


Рис 4.1.3

4.2. Надо скопировать отчеты по выполнению предыдущих лабораторных работ в соответствующие каталоги созданного рабочего пространства. Для этого использую cp и копирую из Downloads в соответствующий каталог созданного рабочего пространства и проверяю с ls:

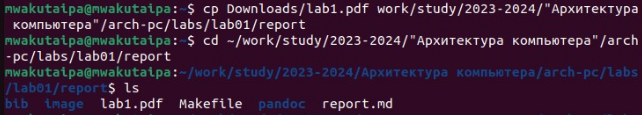


Рис 4.2.0

4.3. Добавляю с помощью команды git add в созданные файлы:

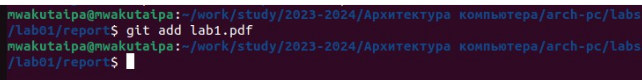


Рис 4.3.0

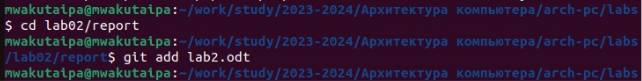


Рис 4.3.1

Сохраняю изменения с помощью git commit:

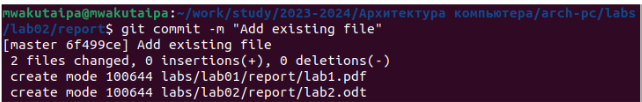


Рис 4.3.2

Отправляю в центральный репозиторий сохраненные изменения командой git push -f origin master:

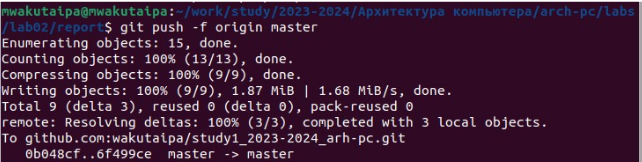


Рис 4.3.3

## 0.5 Выводы

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

## 0.6 Источники

[*Архитектура ЭВМ*](https://esystem.rudn.ru/pluginfile.php/2089082/mod_resource/content/0/Лабораторная%20работа%20№2.%20Система%20контроля%20версий%20Git.pdf)

[*glebradchenko*](https://glebradchenko.susu.ru/courses/bachelor/engineering/2016/SUSU_SE_2016_REP_3_VCS.pdf)