Отчет по лабораторной работе № 3

Архитектура компьютера

Назаров Алексей Михайлович

Содержание

# 1 Цели и задачи

1. Ознакомиться с системой контроля версий GIT.
2. Изучить идеологию применения средств контроля версий.

# 2 Выполнение лабораторной работы

## 2.1 Изменим настройки GIT с помощью git config

(рис. 1)

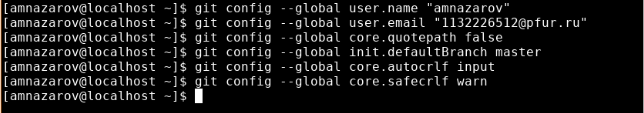


Рис. 1: git config

## 2.2 Создание SSH ключей

Сгенерируем ключи (рис. 2)

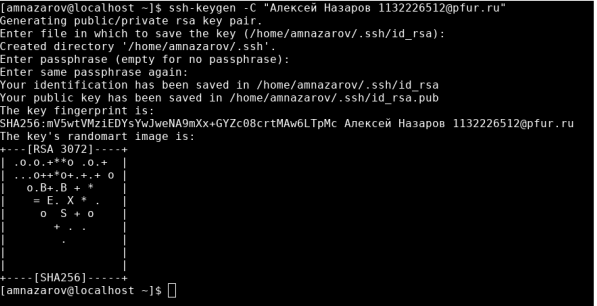


Рис. 2: Создание SSH ключей

Созданные ключи находятся в директории ~/.ssh c.

## 2.3 Скопируем публичный ключ коммандой:

(рис. 3)

Рис. 3: Скопируем публичный ключ

Рис. 3: Скопируем публичный ключ

## 2.4 Добавим публичный ключ на GitHub

Создадим аккаунт на GitHub и добавим Ssh ключ. (рис. 4)

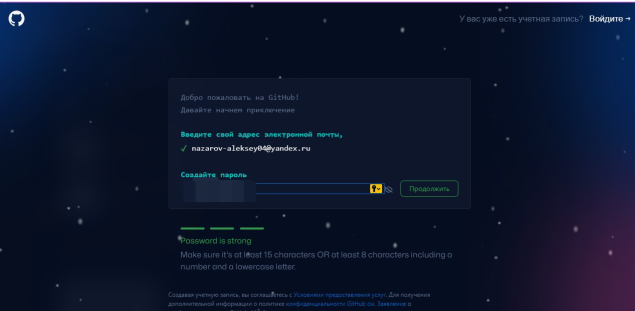


Рис. 4: Окно создания задания логина и пароля на гитхабе

Откроем вкладку SSH and GPG keys, перейдя в настройки. (рис. 5)

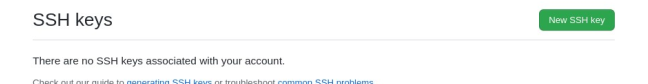


Рис. 5: New SSH key

## 2.5 Добавим публичный ключ

(рис. 6)

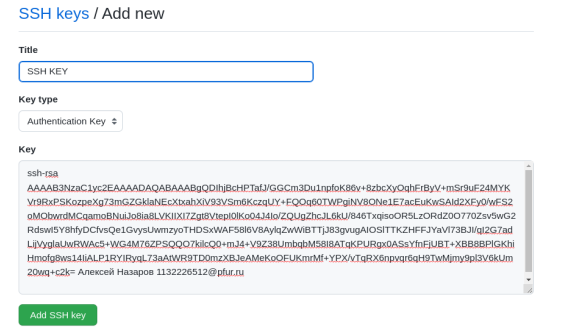


Рис. 6: Добавление нового SSH Ключа

## 2.6 Откроем терминал и создадим рабочее пространство и каталог для предмета «Архитектура компьютера»

(рис. 7)

Рис. 7: Создание рабочего пространствa

Рис. 7: Создание рабочего пространствa

## 2.7 Создадим новый репозиторий из шаблона

(рис. 8)

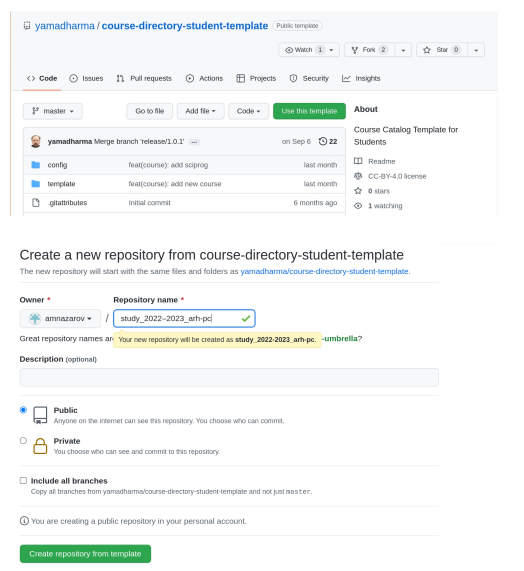


Рис. 8: Создание из шаблона

## 2.8 Перейдем в каталог курса и склонируем репозиторий

(рис. 9)

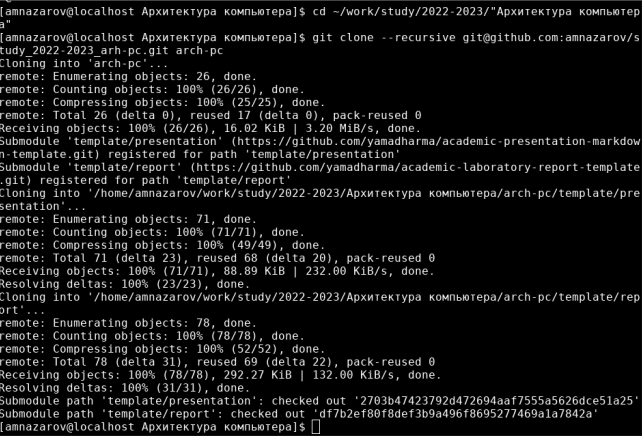


Рис. 9: Клонирование репозитоиия c GitHub

Теперь каталог курса находится в arch-pc.

## 2.9 Настроим содержимое курса

Удалим файл package.json. (рис. 10)

Рис. 10: rm package.json

Рис. 10: rm package.json

Создадим необходимые каталоги. (рис. 11)

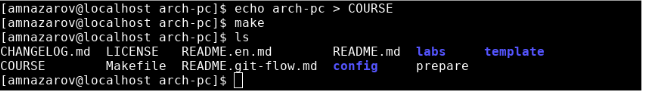


Рис. 11: cоздание файла COURSE и выполнение комманды make

Исполнив комманду ls, мы проверили что package.json удалился, а COURSE создался. Также, после выполнения команды make, создался каталог labs.

## 2.10 Отправим файлы на сервер

(рис. 12), (рис. 13), (рис. 14)



Рис. 12: git add .

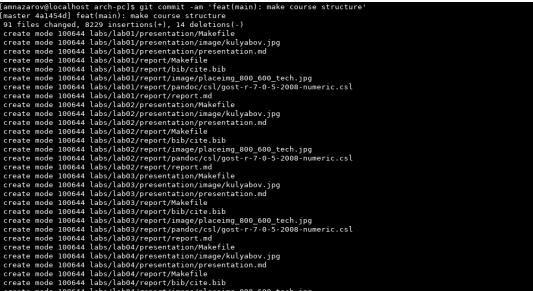


Рис. 13: Git commit

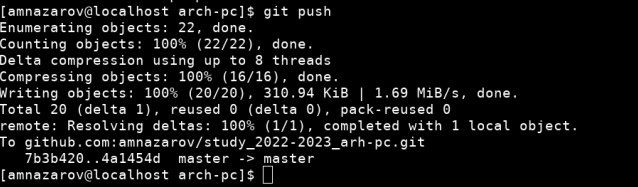


Рис. 14: Git push

## 2.11 Перейдем на страницу репозитория на GitHub и проверим правильность

(рис. 15)

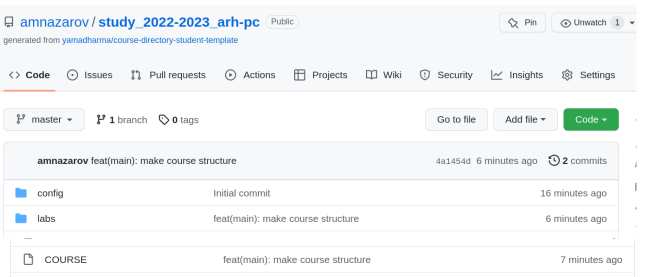


Рис. 15: Страница репозитория на GitHub

Видим, что последние изменения 6 минут назад и появился каталог labs.

# 3 Задания для самостоятельной работы

## 3.1 Создадим отчет по выполнению лабораторной работы в соответствующем каталоге рабочего пространства

(рис. 16)

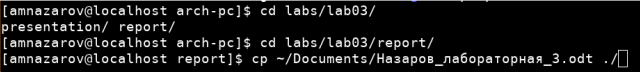


Рис. 16: Копирование третьей лабораторной в рабочее пространство

Я создаю отчет во время выполнения лабораторной работы, поэтому нужно просто его переместить в папку с отчетами.

## 3.2 Скопируем отчеты о предыдущих работах в соответствующий каталог рабочего пространства.

(рис. 17)

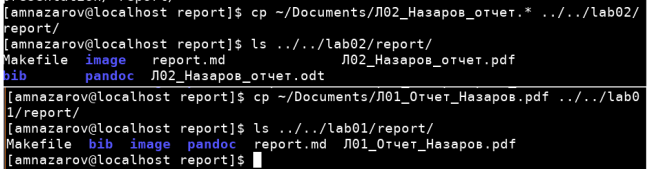


Рис. 17: Kопирование лабораторных 1 и 2 в рабочее пространство

Теперь предыдущие отчеты лежат в папке report каталогов lab02 и lab03. Так как я находился в каталоге lab03, я указазывал относительные пути ../../lab02/report и ../../lab01/report, что бы не писать полные пути.

## 3.3 Загрузим файлы на GitHub

Для этого вернемся из lab03 в arch-pc и выполним комманду git add. (рис. 18)

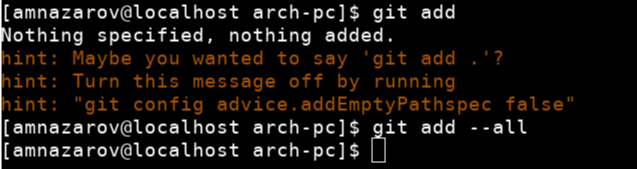


Рис. 18: Комманда git add –all

Я использовал опцию —all, что бы добавить все изменения.

Выполним комманду git commit что бы сохранить изменения. (рис. 19)

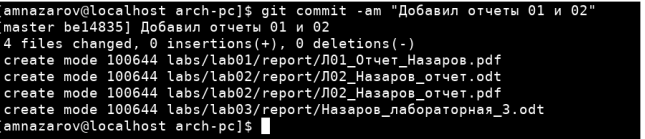


Рис. 19: Коммит с названием «Добавил отчеты 01 и 02»

Загрузим изменения на GitHub. рис. 20)

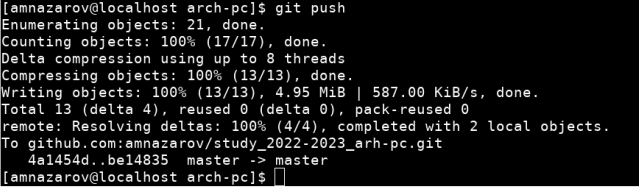


Рис. 20: Создание коммита с сообщением “Add previous lab reports”

## 3.4 Выводы

При проведении лабораторной работы, мы ознакомились с системой контроля версий GIT, изучили идеологию применения VCS, создали репозиторий из шаблона, внесли в него некоторые изменения и добавили свои отчеты.