

Отчёта по лабораторной работе № 2

Система контроля версий Git

Гущина Екатерина Антоновна

Содержание

1 Цель работы	5
2 Выполнение лабораторной работы	6
2.1 Настройка github	6
2.2 Базовая настройка git	6
2.3 Создание SSH-ключа	7
2.4 Создание рабочего пространства и репозитория курса на основе шаблона	8
2.5 Создание репозитория курса на основе шаблона	8
2.6 Настройка каталога курса	9
3 Задание для самостоятельной работы	11
4 Выводы	12

Список иллюстраций

2.1 Настройка автора	6
2.2 Настройка текста	6
2.3 Создание ключей	7
2.4 Создание рабочего каталога	8
2.5 Создание репозитория на основе шаблона	8
2.6 Клонирование созданного репозитория	9
2.7 Создание иерархии рабочего пространства	10
2.8 Отправка изменений на сервер	10

Список таблиц

1 Цель работы

Изучение идеологии и применения средств контроля версий, приобретение практических навыков по работе с системой контроля версий git.

2 Выполнение лабораторной работы

2.1 Настройка github

Создана учетная запись на GitHub по адресу <https://github.com/eaguthina>

2.2 Базовая настройка git

С помощью команды git config произведем базовую настройку git.

Укажем имя и email владельца репозитория:

```
katya@desktop:~$ git config --global user.name "Гущина Екатерина Антоновна"
katya@desktop:~$ git config --global user.email "eaguthina@example.com"
katya@desktop:~$ 
```

Рисунок 2.1: Настройка автора

Настроим utf-8 в выводе сообщений git, зададим имя начальной ветки и параметры autocrlf/safecrlf:

```
katya@desktop:~$ git config --global core.quotepath false
katya@desktop:~$ git config --global init.defaultBranch master
katya@desktop:~$ git config --global core.autocrlf input
katya@desktop:~$ git config --global core.safecrlf warn
katya@desktop:~$ 
```

Рисунок 2.2: Настройка текста

2.3 Создание SSH-ключа

Сгенерируем с помощью утилиты ssh-keygen пару ключей для последующей идентификации пользователя на сервере репозиториев:

```
katya@desktop:~$ ssh-keygen -C "Гущина Екатерина Антоновна eaguthina@example.com"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/katya/.ssh/id_ed25519):
Created directory '/home/katya/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/katya/.ssh/id_ed25519
Your public key has been saved in /home/katya/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:b+K69lZBQJ4hMyGTy+hWxnVOuL3d1IAUfhxaAXHABLМ Гущина Екатерина Антоновна eaguthina@example.com
The key's randomart image is:
+--[ED25519 256]--+
|   o.=+***B*=. |
|   .oo=o*+=.. |
| + o *E.o oo |
| . * . o ... . |
| . o     So + |
| o       ..o . |
| .         ...o |
|             ...o |
|             .o=0 |
+---[SHA256]-----+
katya@desktop:~$ 
```

Рисунок 2.3: Создание ключей

В каталоге `~/ssh` были созданы два ключа: открытый и закрытый.

Сгенерированный открытый ключ (`~/ssh/id_ed25519.pub`) скопируем и добавим для своей учетной записи github, а закрытый добавим локальному ssh

```
katya@desktop:~$ mkdir -p ~/work/study/2025-2026/"Архитектура компьютера"
katya@desktop:~$ cd ~/work/study/2025-2026/Архитектура\ компьютера/
katya@desktop:~/work/study/2025-2026/Архитектура\ компьютера$ 
```

агенту:

Ключи добавлены.

2.4 Создание рабочего пространства и репозитория курса на основе шаблона

Создадим каталог для предмета «Архитектура компьютера»:

```
katya@desktop:~$ cat ~/.ssh/id_ed25519.pub | xclip -sel clip
katya@desktop:~$ ssh-add ~/.ssh/id_ed25519
Enter passphrase for /home/katya/.ssh/id_ed25519:
Identity added: /home/katya/.ssh/id_ed25519 (Гущина Екатерина Антоновна eaguthina@example.com)
katya@desktop:~$ 
```

Рисунок 2.4: Создание рабочего каталога

2.5 Создание репозитория курса на основе шаблона

Перейдем на страницу репозитория с шаблоном курса <https://github.com/yamadharma/course-directory-student-template>. Создадим репозиторий на основе данного шаблона с названием study_2025–2026_arh-pc:

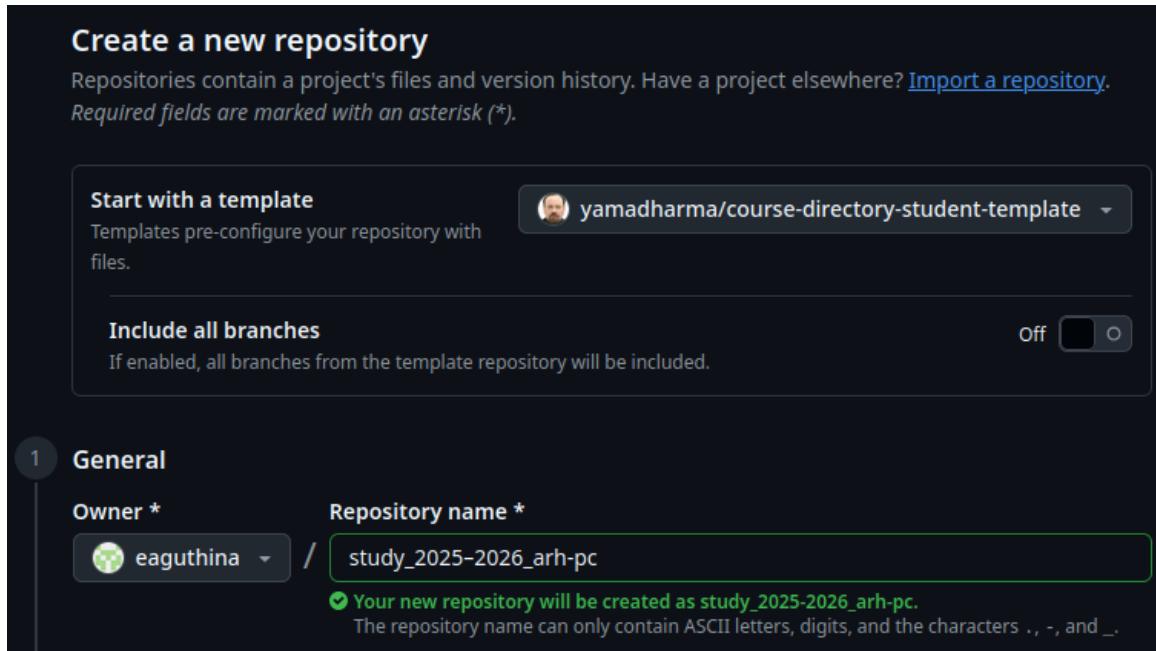


Рисунок 2.5: Создание репозитория на основе шаблона

Клонируем (рекурсивно) созданный выше репозиторий командой git clone:

```
katya@retmasware:~/work/study/2025-2026/Архитектура компьютера$ git clone --recursive git@github.com:eagu
thina/study_2025-2026_arh-pc.git arch-pc
Клонирование в «arch-pc»...
remote: Enumerating objects: 38, done.
remote: Counting objects: 100% (38/38), done.
remote: Compressing objects: 100% (36/36), done.
remote: Total 38 (delta 1), reused 26 (delta 1), pack-reused 0 (from 0)
Получение объектов: 100% (38/38), 23.57 КиБ | 23.57 МиБ/с, готово.
Определение изменений: 100% (1/1), готово.
Подмодуль «template/presentation» (https://github.com/yamadharma/academic-presentation-markdown-template.git) зарегистрирован по пути «template/presentation»
Подмодуль «template/report» (https://github.com/yamadharma/academic-laboratory-report-template.git) зарегистрирован по пути «template/report»
Клонирование в «/home/katya/work/study/2025-2026/Архитектура компьютера/arch-pc/template/presentation»...
remote: Enumerating objects: 219, done.
remote: Counting objects: 100% (219/219), done.
remote: Compressing objects: 100% (151/151), done.
remote: Total 219 (delta 86), reused 189 (delta 56), pack-reused 0 (from 0)
Получение объектов: 100% (219/219), 2.66 МиБ | 4.46 МиБ/с, готово.
Определение изменений: 100% (86/86), готово.
Клонирование в «/home/katya/work/study/2025-2026/Архитектура компьютера/arch-pc/template/report»...
remote: Enumerating objects: 251, done.
remote: Counting objects: 100% (251/251), done.
remote: Compressing objects: 100% (172/172), done.
remote: Total 251 (delta 111), reused 204 (delta 64), pack-reused 0 (from 0)
Получение объектов: 100% (251/251), 775.12 КиБ | 3.34 МиБ/с, готово.
Определение изменений: 100% (111/111), готово.
Submodule path 'template/presentation': checked out '1c93acf9e731bf186384c85de4aff70037314240'
Submodule path 'template/report': checked out '8ee157c58b3362947b1c71492a65d4dc6882d5ad'
katya@retmasware:~/work/study/2025-2026/Архитектура компьютера$
```

Рисунок 2.6: Клонирование созданного репозитория

Теперь в каталоге рабочего пространства содержится наша локальная копия репозитория.

2.6 Настройка каталога курса

Перейдем в каталог курса и создадим необходимые каталоги командой make prepare, добавим изменения командой git add, сохраним изменения командой git commit и отправим их на сервер командой git push:

```
katya@retmasware:~/work/study/2025-2026/Архитектура компьютера/arch-pc$ echo arch-pc > COURSE
katya@retmasware:~/work/study/2025-2026/Архитектура компьютера/arch-pc$ make prepare
katya@retmasware:~/work/study/2025-2026/Архитектура компьютера/arch-pc$ git add .
katya@retmasware:~/work/study/2025-2026/Архитектура компьютера/arch-pc$ git commit -am 'feat(main): make course structure'
[master b244b3c] feat(main): make course structure
 248 files changed, 8698 insertions(+), 225 deletions(-)
```

Рисунок 2.7: Создание иерархии рабочего пространства

```
katya@retmasware:~/work/study/2025-2026/Архитектура компьютера/arch-pc$ git push
Перечисление объектов: 73, готово.
Подсчет объектов: 100% (73/73), готово.
При сжатии изменений используется до 12 потоков
Сжатие объектов: 100% (56/56), готово.
Запись объектов: 100% (70/70), 701.01 КиБ | 4.87 МиБ/с, готово.
Всего 70 (изменений 24), повторно использовано 0 (изменений 0), повторно использовано пакетов
0
remote: Resolving deltas: 100% (24/24), completed with 1 local object.
To github.com:eaguthina/study_2025-2026_arh-pc.git
 8ba3270..b244b3c master -> master
katya@retmasware:~/work/study/2025-2026/Архитектура компьютера/arch-pc$
```

Рисунок 2.8: Отправка изменений на сервер

Иерархия рабочего пространства создана как в локальном репозитории так и на github.

3 Задание для самостоятельной работы

1. Создадим отчет по выполнению лабораторной работы в соответствующем каталоге рабочего пространства (labs/lab02/report): В каталоге labs/lab02/report созданы файлы отчета данной лабораторной работы Л02_Гущина_отчет.docx (данный файл) и Л02_Гущина_отчет.pdf
2. Скопируем отчет по выполнению предыдущей лабораторной работы в соответствующий каталог созданного рабочего пространства: В каталоге labs/lab01/report добавлены файлы отчета лабораторной работы №1 Л01_Гущина_отчет.docx и Л01_Гущина_отчет.pdf.
3. Загрузим файлы на github. Для этого будем использовать команды:

```
git add .  
git commit -am "feat(main): add lab01 and lab02 reports"  
git push
```

После этого файлы с отчетами будут загружены на удаленный репозиторий github.

4 Выводы

Были изучены идеология и применение средств контроля версий, приобрели практические навыки по работе с системой контроля версий git.