

Лабораторная работа №3

Тимур Андреевич Дарижапов

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	11

Список таблиц

Список иллюстраций

3.1	Рисунок 1	7
3.2	Рисунок 2	7
3.3	Рисунок 3	7
3.4	Рисунок 4	8
3.5	Рисунок 5	8
3.6	Рисунок 6	8
3.7	Рисунок 7	8
3.8	Рисунок 8	9
3.9	Рисунок 9	9
3.10	Рисунок 10	9
3.11	Рисунок 11	10
3.12	Рисунок 12	10
3.13	Рисунок 13	10

1 Цель работы

Изучить идеологию и применение средств контроля версий

2 Задание

Сделайте отчёт по предыдущей лабораторной работе в формате Markdown. В качестве отчёта просьба предоставить отчёты в 3 форматах:pdf,dосхимd(в архиве,поскольку он должен содержать скриншоты,Makefile ит.д.)

3 Выполнение лабораторной работы

1.Настройка git

```
[tadarizhapov@tadarizhapov ~]$ git config --global user.name "Timur Darizhapov"
[tadarizhapov@tadarizhapov ~]$ git config --global user.email "timaanddar@mail.ru"
[tadarizhapov@tadarizhapov ~]$ git config --global quotepath false
```

Рис. 3.1: Рисунок 1

Создаём локальный репозиторий

```
[tadarizhapov@tadarizhapov ~]$ cd
[tadarizhapov@tadarizhapov ~]$ mkdir laboratory
[tadarizhapov@tadarizhapov ~]$ cd laboratory
[tadarizhapov@tadarizhapov laboratory]$ git init
Initialized empty Git repository in /home/tadarizhapov/laboratory/.git/
[tadarizhapov@tadarizhapov laboratory]$ █
```

Рис. 3.2: Рисунок 2

```
[tadarizhapov@tadarizhapov laboratory]$ ssh-keygen -C "Timur Darizhapov <timaanddar@mail.ru>"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/tadarizhapov/.ssh/id_rsa): key
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in key.
Your public key has been saved in key.pub.
The key fingerprint is:
SHA256:wawKzD9UgkFbMs3o+54a8hzhDwfXy/A1P6WjleG2oMg Timur Darizhapov <timaanddar@mail.ru>
The key's randomart image is:
+---[RSA 2048]-----+
|  .+=.  .  |
|  .Bo. o  |
|  .o . =   |
|  o.. + o  |
|  *. = S . o . . |
|  ..0 . + o + = |
|  . +. + . X   |
|  + o. + . . = + |
|  +oo E . . . |
+---[SHA256]-----+
```

Рис. 3.3: Рисунок 3

Регистрируемся на github.com. Переходим в Settings–SSHkeys. Вставляем в поле наш публичный ключ с помощью команды `cat ~/.ssh/id_rsa.pub`

SSH keys / Add new

Title

Key

Begins with 'ssh-rsa', 'ecdsa-sha2-nistp256', 'ecdsa-sha2-nistp384', 'ecdsa-sha2-nistp521', or 'ssh-ed25519'

Рис. 3.4: Рисунок 4

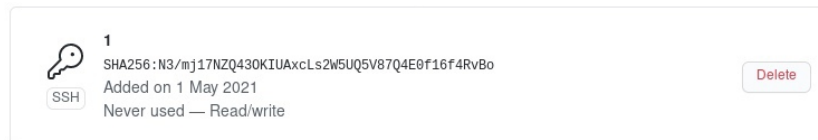


Рис. 3.5: Рисунок 5

2. Создаём репозиторий на сайте github.com. Делаем его публичным.

Owner * / Repository name *

Great repository names are short and memorable. Need inspiration? How about **redesigned-octo-engine**?

Description (optional)

☒ **Public**
Anyone on the internet can see this repository. You choose who can commit.

☐ **Private**
You choose who can see and commit to this repository.

Рис. 3.6: Рисунок 6

Создаём README.md. Печатаем в нём описание. Делаем первый commit.

```
[tadarizhapov@tadarizhapov laboratory]$ echo "#Лабораторные работы" >> README.md
[tadarizhapov@tadarizhapov laboratory]$ git add README.md
[tadarizhapov@tadarizhapov laboratory]$ git commit -m "my first commit"
[master (root-commit) cb69ee0] my first commit
1 file changed, 1 insertion(+)
create mode 100644 README.md
[tadarizhapov@tadarizhapov laboratory]$ git status
# On branch master
# Untracked files:
#   (use "git add <file>..." to include in what will be committed)
#
#       key
#       key.pub
nothing added to commit but untracked files present (use "git add" to track)
```

Рис. 3.7: Рисунок 7

Отправляем на github.com.

```
[tadarizhapov@tadarizhapov laboratory]$ git push -u origin master
Username for 'https://github.com': h3avydirtysoul
Password for 'https://h3avydirtysoul@github.com':
Counting objects: 3, done.
Writing objects: 100% (3/3), 252 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:   https://github.com/h3avydirtysoul/tadarizhapov/pull/new/master
remote:
To https://github.com/h3avydirtysoul/tadarizhapov.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from origin.
```

Рис. 3.8: Рисунок 8

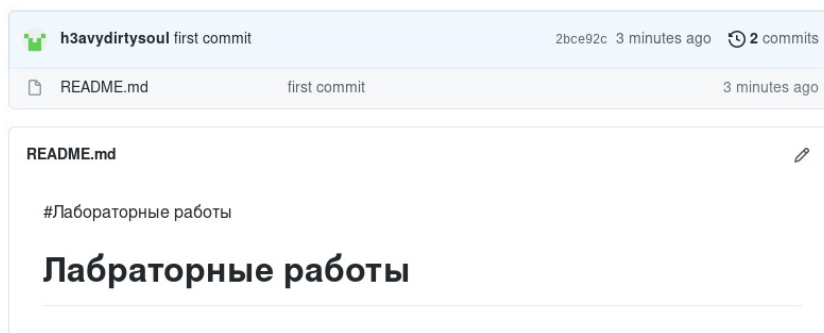


Рис. 3.9: Рисунок 9

3.Добавляем файл лицензии и шаблон игнорируемых файлов.

```
[tadarizhapov@tadarizhapov laboratory]$ wget https://creativecommons.org/licenses/by/4.0/legalcode.txt
--2021-05-01 10:21:23-- https://creativecommons.org/licenses/by/4.0/legalcode.txt
Распознаётся creativecommons.org (creativecommons.org)... 172.67.34.140, 104.20.151.16, 104.20.150.16, ...
Подключение к creativecommons.org (creativecommons.org)[172.67.34.140]:443... соединение установлено.
HTTP-запрос отправлен. Ожидание ответа... 200 OK
Длина: нет данных [text/plain]
Сохранение в: «legalcode.txt»

[ <=> ] 18 657 --.-K/s за 0s
2021-05-01 10:21:24 (73,6 MB/s) - «legalcode.txt» сохранён [18657]
```

Рис. 3.10: Рисунок 10

Добавляем два новых файла, совершаем commitи отправляем на сайт.

```
[tadarizhapov@tadarizhapov laboratory]$ .gitignore
bash: .gitignore: команда не найдена...
[tadarizhapov@tadarizhapov laboratory]$ curl -L -s https://www.gitignore.io/api/c >> .gitignore
[tadarizhapov@tadarizhapov laboratory]$ git add .
[tadarizhapov@tadarizhapov laboratory]$ git status
# On branch master
# Changes to be committed:
#   (use "git reset HEAD <file>..." to unstage)
#
#       new file:   .gitignore
#       new file:   key
#       new file:   key.pub
#       new file:   legalcode.txt
#
```

Рис. 3.11: Рисунок 11

```
[tadarizhapov@tadarizhapov laboratory]$ git push
warning: push.default is unset; its implicit value is changing in
Git 2.0 from 'matching' to 'simple'. To squelch this message
and maintain the current behavior after the default changes, use:

    git config --global push.default matching

To squelch this message and adopt the new behavior now, use:

    git config --global push.default simple

See 'git help config' and search for 'push.default' for further information.
(the 'simple' mode was introduced in Git 1.7.11. Use the similar mode
'current' instead of 'simple' if you sometimes use older versions of Git)

Username for 'https://github.com': h3avydirtysoul
Password for 'https://h3avydirtysoul@github.com':
Counting objects: 7, done.
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 8.19 KiB | 0 bytes/s, done.
Total 6 (delta 0), reused 0 (delta 0)
To https://github.com/h3avydirtysoul/tadarizhapov.git
 2bce92c..e5b3ef5 master -> master
```

Рис. 3.12: Рисунок 12

4. В первой лабораторной работе было сказано установить образ операционной системы CentOS. В лабораторной работе есть команды только для Gentoo и Ubuntu. Ни одна из этих команд не помогла мне установить gitflow. Посмотрев в интернете, решения этой проблемы я не нашёл.

```
[tadarizhapov@tadarizhapov laboratory]$ emerge dev-vcs/git-flow
bash: emerge: команда не найдена...
Аналогичная команда: 'merge'
[tadarizhapov@tadarizhapov laboratory]$ apt-get install git-flow
bash: apt-get: команда не найдена...
[tadarizhapov@tadarizhapov laboratory]$ sudo apt-get install git-flow
[sudo] пароль для tadarizhapov:
sudo: apt: command not found
[tadarizhapov@tadarizhapov laboratory]$ apt-get install git-flow
bash: apt-get: команда не найдена...
```

Рис. 3.13: Рисунок 13

4 Выводы

Я изучил применение средств контроля версий и их идеологию, научился на начальном уровне пользоваться git.