

Лабораторная работа 1

Агоннудэ Месседэ Мишель НКНбд-01-19

10 февраля, 2022, Москва, Россия

Российский Университет Дружбы Народов

Цели и задачи работы

Цель лабораторной работы

Целью данной работы является изучение идеологии и применения средств контроля версий.

Задачи лабораторной работы

1. Создать учетную запись на github.com
2. Настроить репозиторий
3. Изучить механизм управления версиями

Процесс выполнения лабораторной работы


Создаем учетную запись на github.com и репозиторий

Create a new repository


A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner ^{*}

Repository name ^{*}


 MishelMessed

 /


MatMod 

Great repository names are short and memorable. Need inspiration? How about [expert-guide](#)?

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.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☐ **Add a README file**

This is where you can write a long description for your project. [Learn more.](#)

☐ **Add .gitignore**

Choose which files not to track from a list of templates. [Learn more.](#)

☐ **Choose a license**

A license tells others what they can and can't do with your code. [Learn more.](#)

Create repository

Figure 1: Создание репозитория

Инициализируем локальный репозиторий

```
PS C:\Users\User\Desktop\University\labs> git init
Initialized empty Git repository in C:/Users/User/Desktop/University/labs/.git/
PS C:\Users\User\Desktop\University\labs> echo "# лабораторные работы" >> README.md
PS C:\Users\User\Desktop\University\labs> git add README.md
PS C:\Users\User\Desktop\University\labs> echo "# лабораторные работы" >> README.md
PS C:\Users\User\Desktop\University\labs> git add README.md
PS C:\Users\User\Desktop\University\labs>
```

Figure 2: Инициализация репозитория

Создаем SSH-ключ

```
PS C:\Users\User\Desktop\University\labs> git add README.md
PS C:\Users\User\Desktop\University\labs> git config --global user.name MisheMessed
PS C:\Users\User\Desktop\University\labs> git config --global user.email "1032189125@pfur.ru"
PS C:\Users\User\Desktop\University\labs> git commit -m "first commit"
[master (root-commit) ac51e1] first commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 README.md
PS C:\Users\User\Desktop\University\labs> ssh-keygen -C "MisheMessed 1032189125@pfur.ru"
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\User/.ssh/id_rsa):
Created directory 'C:\Users\User/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\User/.ssh/id_rsa.
Your public key has been saved in C:\Users\User/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:/cZ4/F8u042MUv8W6RZEint9ggNH7Ye1RbwDAT4oEU MisheMessed 1032189125@pfur.ru
The key's randomart image is:
----[RSA 2048]-----
      .E*..+.*+
      O..+0+==
      . .o+==+
      . .oo==*g
      S . .oo
      * O
      . B +
      O .o
      +O
-----[SHA256]-----
PS C:\Users\User\Desktop\University\labs> cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDAgk03o3jZ9RohCDswLq6CtwThxp5W3nx4ZLrypttb0ryPKLAjdXr1ptjHPU0447+DnCLH0gph2mJg
e70NF39a1rW1+Lhry+CWj3FmJqLCAWkt3AkQ64SdnJTEopqpgNB00amYAj0DhBVkVFTurodTtco4no+nozyu7q5M0rRb81x3oqPhwL0QkzTVhpm8m2s
PAoL71nfWf2E3R+j1L50wCj1063pUEAug+K7Z/sfXkpbX0X6kgmsqs60onTAP1WJ1qkVP2evPXIzCTVd1t1FwVJTP3mepc3ipMERyqA5hsJyVh1sT7b
C8RVP67Fxy13T91s0z2x MisheMessed 1032189125@pfur.ru
PS C:\Users\User\Desktop\University\labs>
```

Figure 3: Создание SSH-ключа

Создаем SSH-ключ

SSH keys / Add new

Title

Key

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDnRko5Q3o3JjZ9RoHcDSWILqBGtwThXpSW3nxAzLrYpttbQrvPKLAjdXripMjHP
UO447+DNCLM0gPm2mjqe7ONFX9alrMi+LhrY+CwhJfnUqnLCAaWkt3aMqG4SDmJIeopqqgGN8ODamYA/QDhBvkVKT
urodTtcO4no+mozyu7gxMOrRb8lx3oqPhwHLOQkzTYhpm8mZsPAodL/lmfWR2F3R+jtL5QWCjIO6ZpUeAuG+K7Z/Sfxxhp
BxOX6KqMsqs5OOonTAPIUHJlqKVP2eVPXlzCTVdltiFWvJTP3mepcCipKMERYQASHsjaYh1isT7bGRKPYG7fy1T9lvs8Zx
MishelMessed 1032189125@pfur.ru
```

Add SSH key

Figure 4: Добавление ключа на github.com

Загружаем служебные файлы

```
PS C:\Users\User\Desktop\University\Tabs>
PS C:\Users\User\Desktop\University\Tabs> git remote add origin git@github.com:MissheMessed/MatMod.git
PS C:\Users\User\Desktop\University\Tabs> wget https://creativecommons.org/licenses/by/4.0/legalcode.txt -O LICENSE
PS C:\Users\User\Desktop\University\Tabs> wget https://www.toptal.com/developers/gitignore/api/python -O .gitignore
PS C:\Users\User\Desktop\University\Tabs> git add .
warning: LF will be replaced by CRLF in .gitignore.
The file will have its original line endings in your working directory
PS C:\Users\User\Desktop\University\Tabs> git commit -am 'add license'
warning: LF will be replaced by CRLF in .gitignore.
The file will have its original line endings in your working directory
warning: LF will be replaced by CRLF in LICENSE.
The file will have its original line endings in your working directory
[master 256327d] add license
 2 files changed, 555 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 LICENSE
PS C:\Users\User\Desktop\University\Tabs> git push -u origin master
The authenticity of host 'github.com (140.82.121.3)' can't be established.
ED25519 key fingerprint is SHA256:+D13wvV6TuJJhbp2isF/zLDADzPMSvHdkr40uCOq9.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 712 KiB | 2.57 MiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:MissheMessed/MatMod.git
 * [new branch] master -> master
branch 'master' set up to track 'origin/master'.
PS C:\Users\User\Desktop\University\Tabs> git push
Everything up-to-date
PS C:\Users\User\Desktop\University\Tabs>
```

Figure 5: Загрузка файлов лицензии и gitignore

Использование системы управления версиями

```
PS C:\Users\User\Desktop\University\labs> git flow release start 1.0.0
Switched to a new branch 'release/1.0.0'

Summary of actions:
- A new branch 'release/1.0.0' was created, based on 'develop'
- You are now on branch 'release/1.0.0'

Follow-up actions:
- Bump the version number now!
- Start committing last-minute fixes in preparing your release
- When done, run:

    git flow release finish '1.0.0'

PS C:\Users\User\Desktop\University\labs> echo "1.0.0" >> VERSION
PS C:\Users\User\Desktop\University\labs> git add .
PS C:\Users\User\Desktop\University\labs> git commit -am "chore(main): add version"
[release/1.0.0 ce85d79] chore(main): add version
1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 VERSION
PS C:\Users\User\Desktop\University\labs> git flow release finish -m "ver 1" 1.0.0
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
Merge made by the 'ort' strategy.
VERSION | Bin 0 -> 10 bytes
1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 VERSION
Already on 'master'
Your branch is ahead of 'origin/master' by 2 commits.
(use "git push" to publish your local commits)
Switched to branch 'develop'
Merge made by the 'ort' strategy.
VERSION | Bin 0 -> 10 bytes
1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 VERSION
Deleted branch release/1.0.0 (was ce85d79).

Summary of actions:
- Release branch 'release/1.0.0' has been merged into 'master'
- The release was tagged 'v1.0.0'
- Release tag 'v1.0.0' has been back-merged into 'develop'
- Release branch 'release/1.0.0' has been locally deleted
- You are now on branch 'develop'

PS C:\Users\User\Desktop\University\labs>
```

Figure 6: Инициализация git-flow и создание релиза

Использование системы управления версиями

```
PS C:\Users\User\Desktop\University\labs> git push --all
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 479 bytes | 239.00 KiB/s, done.
Total 5 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (3/3), completed with 1 local object.
To github.com:MichelMessed/MatMod.git
 * 3ed1ba6..4e57295 master -> master
 * [new branch]      develop -> develop
PS C:\Users\User\Desktop\University\labs> git push --tags
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 164 bytes | 164.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:MichelMessed/MatMod.git
 * [new tag]         v1.0.0 -> v1.0.0
PS C:\Users\User\Desktop\University\labs>
```

Figure 7: Отправка изменений в сетевой репозиторий

Выполним объединение веток

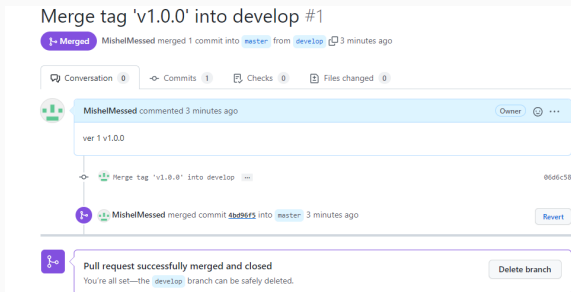


Figure 8: Объединение веток в сетевом репозитории

Выводы по проделанной работе

Мы приобрели практические навыки работы с системой контроля версий git и создали свой репозиторий