Лабораторная работа 1

Нгуен Тхай Зыонг НПИбд-02-19

11 февраля, 2022, Москва, Россия

Российский Университет Дружбы Народов

Цели и задачи работы

Цель лабораторной работы

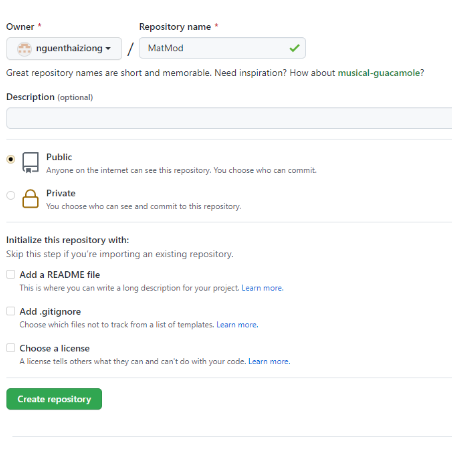
Целью данной работы является изучение идеологии и применения средств контроля версий.

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



Owner * nguenthaiziong / Repository name * MatMod ✓

Great repository names are short and memorable. Need inspiration? How about musical-guacamole?

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.

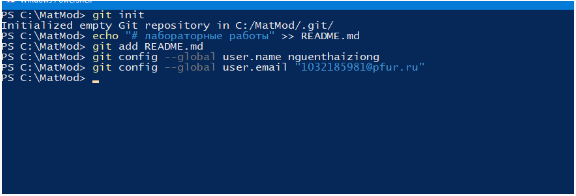
□ Add a README file
This is where you can write a long description for your project. Learn more.

□ 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.

Figure 1: Создание репозитория

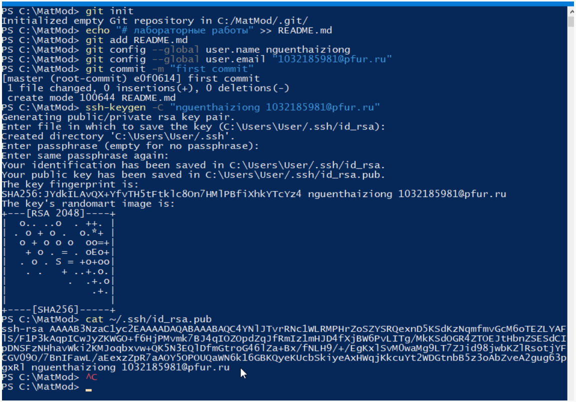
Инициализируем локальный репозиторий



```
PS C:\MatMod> git init
Initialized empty Git repository in C:/MatMod/.git/
PS C:\MatMod> echo "# лабораторные работы" >> README.md
PS C:\MatMod> git add README.md
PS C:\MatMod> git config --global user.name nguenthaiziong
PS C:\MatMod> git config --global user.email "1032185981@pfur.ru"
PS C:\MatMod> _
```

Figure 2: Инициализация репозитория

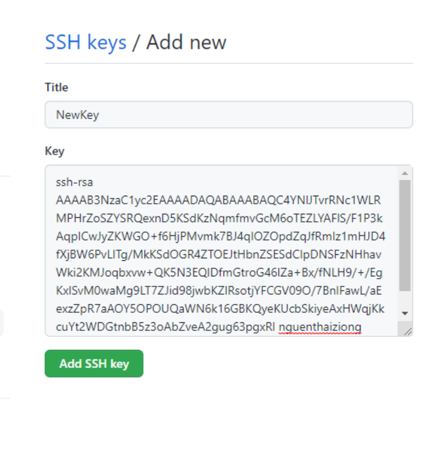
Создаем SSH-ключ



```
PS C:\MatMod> git init
Initialized empty Git repository in C:/MatMod/.git/
PS C:\MatMod> echo "# лабораторные работы" >> README.md
PS C:\MatMod> git add README.md
PS C:\MatMod> git config --global user.name nguenthaiziong
PS C:\MatMod> git config --global user.email "1032185981@pfur.ru"
PS C:\MatMod> git commit -m "first commit"
[master (root-commit) e0f0614] first commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 README.md
PS C:\MatMod> ssh-keygen -C "nguenthaiziong 1032185981@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:JYdkILAvQX+YfvTH5tFtklc8On7HMlPBfiXhkYTcYz4 nguenthaiziong 1032185981@pfur.ru
The key's randomart image is:
+---[RSA 2048]----+
|  o.. ..o  . ++. |
| . o + o .  o.*+ |
|  o + o o o  oo=+|
|   + o . = . oEo+|
|  . o . S = +o+oo|
|   . .   + ..+.o.|
|          .  .+.o|
|              .+.|
|                 |
+----[SHA256]-----+
PS C:\MatMod> cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQC4YNlJTvrRNc1WLRMPHrZoSZYSRQexnD5KSdKzNqmfmvGcM6oTEZLYAF
lS/F1P3kAqpICwJyZKWGO+f6HjPMvmk7BJ4qIOZOpdZqJfRmIz1mHJD4fXjBW6PvLITg/MkKSdOGR4ZTOEJtHbnZSESdCI
pDNSFzNHhavWki2KMJoqbxvw+QK5N3EQlDfmGtroG46lZa+Bx/fNLH9/+/EgKxlSvM0waMg9LT7ZJid98jwbKZlRsotjYF
CGV09O/7BnIFawL/aEexzZpR7aAOY5OPOUQaWN6k16GBKQyeKUcbSkiyeAxHWqjKkcuYt2WDGtnbB5z3oAbZveA2gug63p
gxRl nguenthaiziong 1032185981@pfur.ru
PS C:\MatMod> ^C
PS C:\MatMod> _
```

Figure 3: Создание SSH-ключа

Создаем SSH-ключ



SSH keys / Add new

Title

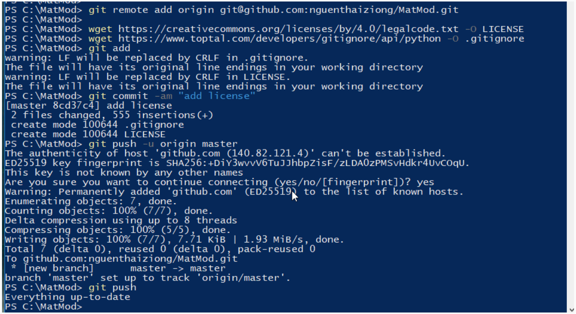
NewKey

Key

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQC4YNIJTvrRNc1WLR
MPHrZoSZYSRQexnD5KSdKzNqmfmvGcM6oTEZLYAFIS/F1P3k
AqplCwJyZKWGO+f6HjPMvmk7BJ4qIOZOpdZqJfRmlz1mHJD4
fXjBW6PvLITg/MkKSdOGR4ZTOEJtHbnZSESdClpDNSFzNHhav
Wki2KMJoqbxvw+QK5N3EQIDfmGtroG46IZa+Bx/fNLH9/+/Eg
KxlSvM0waMg9LT7ZJid98jwbKZIRsotjYFCGV09O/7BnIFawL/aE
exzZpR7aAOY5OPOUQaWN6k16GBKQyeKUcbSkiyeAxHWqjKk
cuYt2WDGtnbB5z3oAbZveA2gug63pgxRl nguenthaizìong
```

Figure 4: Добавление ключа на github.com

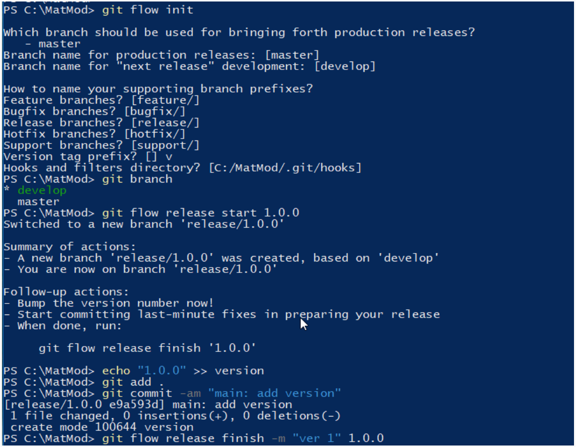
Загружаем служебные файлы



```
PS C:\MatMod>
PS C:\MatMod> git remote add origin git@github.com:nguenthaiziong/MatMod.git
PS C:\MatMod>
PS C:\MatMod> wget https://creativecommons.org/licenses/by/4.0/legalcode.txt -O LICENSE
PS C:\MatMod> wget https://www.toptal.com/developers/gitignore/api/python -O .gitignore
PS C:\MatMod> git add .
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
PS C:\MatMod> git commit -am "add license"
[master 8cd37c4] add license
 2 files changed, 555 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 LICENSE
PS C:\MatMod> git push -u origin master
The authenticity of host 'github.com (140.82.121.4)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvvV6TuJJhbpZisF/zLDA0zPMSvHdkr4UvCOqU.
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% (5/5), done.
Writing objects: 100% (7/7), 7.71 KiB | 1.93 MiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:nguenthaiziong/MatMod.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
PS C:\MatMod> git push
Everything up-to-date
PS C:\MatMod>
```

Figure 5: Загрузка файлов лицензии и gitignore

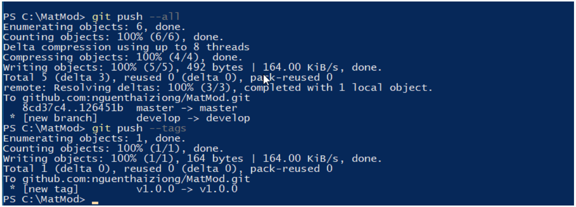
Использование системы управления версиями



```
PS C:\MatMod> git flow init

Which branch should be used for bringing forth production releases?
   - master
Branch name for production releases: [master]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? [] v
Hooks and filters directory? [C:/MatMod/.git/hooks]
PS C:\MatMod> git branch
* develop
  master
PS C:\MatMod> 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:\MatMod> echo "1.0.0" >> version
PS C:\MatMod> git add .
PS C:\MatMod> git commit -am "main: add version"
[release/1.0.0 e9a593d] main: add version
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 version
PS C:\MatMod> git flow release finish -m "ver 1" 1.0.0
```

Figure 6: Инициализация git-flow и создание релиза

Использование системы управления версиями



```
PS C:\MatMod> 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), 492 bytes | 164.00 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:nguenthaizIong/MatMod.git
   8cd37c4..126451b  master -> master
 * [new branch]      develop -> develop
PS C:\MatMod> 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:nguenthaizIong/MatMod.git
 * [new tag]         v1.0.0 -> v1.0.0
PS C:\MatMod> _
```

Figure 7: Отправка изменений в сетевой репозиторий

Выводы по проделанной работе

Вывод

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