

# **Отчёт по лабораторной работе №1**

**Система контроля версий Git**

Стаменкович Огнен НПИбд-02-19

# Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
3	Вывод	8
	Список литературы	9

# List of Figures

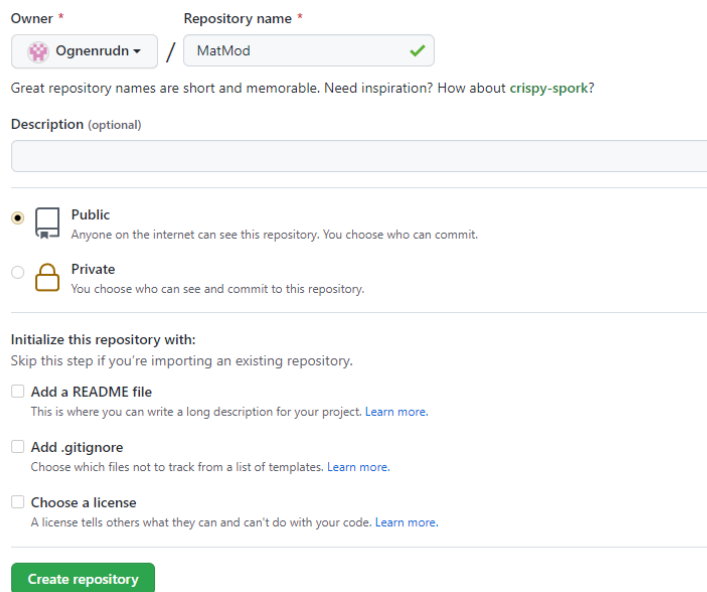
2.1	Создание репозитория . . . . .	5
2.2	Инициализация репозитория . . . . .	5
2.3	Создание SSH-ключа . . . . .	6
2.4	Добавление ключа на github.com . . . . .	6
2.5	Загрузка файлов . . . . .	6
2.6	Инициализация git-flow и начало релиза . . . . .	7
2.7	Завершение релиза и отправка изменений в сетевой репозиторий	7

# 1 Цель работы

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

## 2 Выполнение лабораторной работы

Создаем учетную запись на github.com и репозиторий



Owner \* Repository name \*

Ognerudn / MatMod ✓

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

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 2.1: Создание репозитория

Инициализируем локальный репозиторий и создаю в нем файл README.md

```
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>
```

Figure 2.2: Инициализация репозитория

Создаем SSH-ключ и прописываем его в настройках на github.com

```

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 Ognenrudn
PS C:\MatMod> git config --global user.email "1032195215@pfur.ru"
PS C:\MatMod> git commit -m "first commit"
[master (root-commit) 3486b56] first commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 README.md
PS C:\MatMod> ssh-keygen -s "Ognenrudn 1032195215@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:jgs5Tgch8jouAWLLUGxfz55/joqP88/v41281yzps Ognenrudn 1032195215@pfur.ru
The key's randomart image is:
+---[RSA 2048]-----
+..o.
+o+o
+o+..
+..+..
+.. 0 . S
+.. = + o
+.. * o . . o .
+.. * o . . + B
+o+..+.. o E:
+---[SHA256]-----
PS C:\MatMod> cat -f .ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDAWD/bfbuo/4YNfCK5prxjjoxxgaXumV/inQMlx8HeB2O4VDN1TYngQbXF1av
2eMgHjtkoH+3wJrtUVTOKUvY86+IRnwKNjhjpi6Lklwe7etbKYkV+jeghS9ynO7jKs6cpfuPO0ienPQ9A/Y7Arbosc19188Hf
TJCXoAwMwqTVTMLTwCl0QtydOWmHSHZgHyEDdVK14MCDJ7z7zsdCjv2/KyYML007DrFXCj/wjKlbK9t53wUPHSue9/HoSQ
+AzfYLAJZS6envMhQWKK4mR23nN9VCq4hmsnL0xSSuswGN+oZCtyn63sj8MPKdQSe9HtHCja6lQhLSqk6g4Lx
Ognenrudn 1032195215@pfur.ru
PS C:\MatMod>

```

Figure 2.3: Создание SSH-ключа

## SSH keys / Add new

Title

MatMod

Key

ssh-rsa  
AAAAAB3NzaC1yc2EAAAADAQABAAQDAWD/bfbuo/4YNfCK5prxjjoxxgaXumV/inQMlx8HeB2O4VDN1TYngQbXF1av  
2eMgHjtkoH+3wJrtUVTOKUvY86+IRnwKNjhjpi6Lklwe7etbKYkV+jeghS9ynO7jKs6cpfuPO0ienPQ9A/Y7Arbosc19188Hf  
TJCXoAwMwqTVTMLTwCl0QtydOWmHSHZgHyEDdVK14MCDJ7z7zsdCjv2/KyYML007DrFXCj/wjKlbK9t53wUPHSue9/HoSQ  
+AzfYLAJZS6envMhQWKK4mR23nN9VCq4hmsnL0xSSuswGN+oZCtyn63sj8MPKdQSe9HtHCja6lQhLSqk6g4Lx  
Ognenrudn 1032195215@pfur.ru

Add SSH key

Figure 2.4: Добавление ключа на github.com

Загружаем файлы лицензионного соглашения и gitignore. Отправляем все файлы в сетевой репозиторий.

```

PS C:\MatMod> git remote add origin git@github.com:Ognenrudn/MatMod.git
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 -m "add license"
[master 0d2cc38] 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 (140.82.121.3)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvV6TujzHbP2isF/zLDA0zrPM5vHdK4Uvc0qU.
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 | 1.54 MiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:Ognenrudn/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 2.5: Загрузка файлов

Использование системы управления версиями. Создаем ветку, начинаем и завершаем в ней релиз.

```
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 4681280] 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
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
Merge made by the 'ort' strategy.
 version | Bin 0 -> 16 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.
```

Figure 2.6: Инициализация git-flow и начало релиза

```
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), 468 bytes | 234.00 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:Ognenrudn/MatMod.git
 * [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), 161 bytes | 161.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:Ognenrudn/MatMod.git
 * [new tag]         v1.0.0 -> v1.0.0
PS C:\MatMod>
```

Figure 2.7: Завершение релиза и отправка изменений в сетевой репозиторий

## 3 Вывод

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



# Список литературы

1. Git для новичков
2. Основы Git
3. Руководство по оформлению Markdown файлов