

# **Отчёт по лабораторной работе №1**

**Система контроля версий Git**

Седов Никита Сергеевич НПИбд-01-19

# Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
3	Вывод	9
	Список литературы	10

# List of Figures

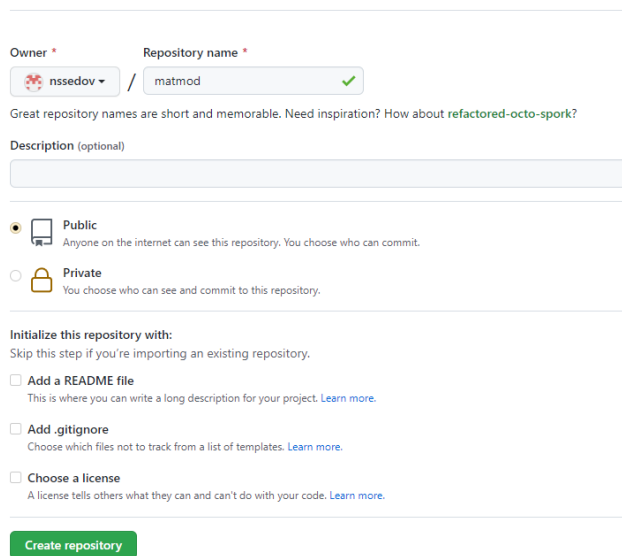
2.1	Создание репозитория . . . . .	5
2.2	Инициализация репозитория . . . . .	5
2.3	Создание SSH-ключа . . . . .	6
2.4	Добавление ключа на github.com . . . . .	6
2.5	Загрузка файлов . . . . .	7
2.6	Инициализация git-flow и начало релиза . . . . .	7
2.7	Завершение релиза и отправка изменений в сетевой репозиторий . . . . .	8
2.8	Объединение веток в сетевом репозитории . . . . .	8

# 1 Цель работы

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

## 2 Выполнение лабораторной работы

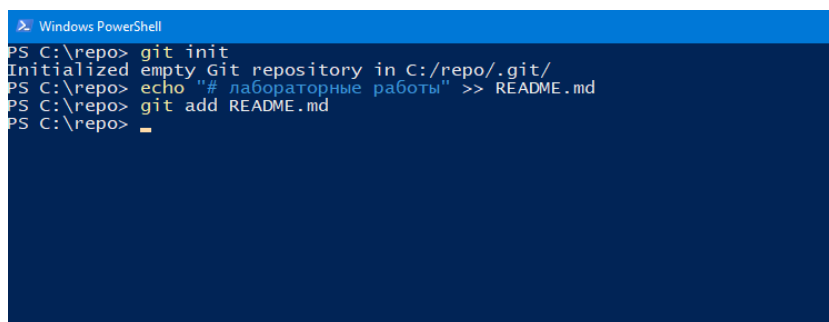
Создаем учетную запись на github.com и репозиторий



The screenshot shows the GitHub 'Create new repository' page. At the top, there are two input fields: 'Owner' with a dropdown menu showing 'nssedov' and a GitHub icon, and 'Repository name' with the text 'matmod' and a green checkmark. Below these fields is a hint: 'Great repository names are short and memorable. Need inspiration? How about [refactored-octo-spork](#)?'. Underneath is a 'Description (optional)' text area. Further down, there are two radio button options for visibility: 'Public' (selected) with the subtext 'Anyone on the internet can see this repository. You choose who can commit.' and 'Private' with the subtext 'You choose who can see and commit to this repository.'. Below the visibility options is a section titled 'Initialize this repository with:' followed by the instruction 'Skip this step if you're importing an existing repository.' and three checkboxes: 'Add a README file' (with subtext 'This is where you can write a long description for your project. [Learn more.](#)'), 'Add .gitignore' (with subtext 'Choose which files not to track from a list of templates. [Learn more.](#)'), and 'Choose a license' (with subtext 'A license tells others what they can and can't do with your code. [Learn more.](#)'). At the bottom of the form is a green 'Create repository' button.

Figure 2.1: Создание репозитория

Инициализируем локальный репозиторий и создаю в нем файл README.md



```
PS C:\repo> git init
Initialized empty Git repository in C:/repo/.git/
PS C:\repo> echo "# лабораторные работы" >> README.md
PS C:\repo> git add README.md
PS C:\repo>
```

Figure 2.2: Инициализация репозитория

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

```
Windows PowerShell
PS C:\repo> git config --global user.name nssedov
PS C:\repo> git config --global user.email "1032190616@pfur.ru"
PS C:\repo> git commit -m "first commit"
[master (root-commit) e5e4663] first commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 README.md
PS C:\repo>
PS C:\repo> ssh-keygen -C "nssedov 1032190616@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:ROxAKBkamRcMfo+FE8H1sBBYaJRkDRNp1njfnkIgLNA nssedov 1032190616@pfur.ru
The key's randomart image is:
+---[RSA 2048]-----+
|X#B+...|
|0%#B...|
|*oo*oo.o.|
|. =o.o.|
|...S|
|.o|
|_|
+---[SHA256]-----+
PS C:\repo> cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDclgdnPxYbVv1f9MgwpfP7lZu1bSyE4Kmt3fmIcVw1QzE8VbPh1rIwTGuMlw4t0SeBGzNKzSVYqrWou+IsYU1VW/kbcolU4jKF9rRVc4DQyY4QzSsBK3K3gPTD2Aro8RDKEXR12IPya0qdkDe8SuUddyZmeRwF10DIDjx9dwZBfmmZ6qIG1cDDtz/+9140LK3P4tkzQN6Xng7qSFyLAKmvOrFSBXBNkafRaHQLDZp2fUowZBzyeE1DsmeH24SKEK2LHMRcIGWqntLrt+UGZcfuG6Re9GOGw5buWxqPLd4PdBoTZB+PsLScpd1d5eVE3yurQ736TgkUfW82HyOQk7nssedov 1032190616@pfur.ru
PS C:\repo>
```

Figure 2.3: Создание SSH-ключа

SSH keys / Add new

Title

ssh

Key

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDclgdnPxYbVv1f9MgwpfP7lZu1bSyE4Kmt3fmIcVw1QzE8VbPh1rIwTGuMlw4t0SeBGzNKzSVYqrWou+IsYU1VW/kbcolU4jKF9rRVc4DQyY4QzSsBK3K3gPTD2Aro8RDKEXR12IPya0qdkDe8SuUddyZmeRwF10DIDjx9dwZBfmmZ6qIG1cDDtz/+9140LK3P4tkzQN6Xng7qSFyLAKmvOrFSBXBNkafRaHQLDZp2fUowZBzyeE1DsmeH24SKEK2LHMRcIGWqntLrt+UGZcfuG6Re9GOGw5buWxqPLd4PdBoTZB+PsLScpd1d5eVE3yurQ736TgkUfW82HyOQk7
nssedov 1032190616@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:\repo> git remote add origin git@github.com:nssedov/matmod.git
PS C:\repo>
PS C:\repo> wget https://creativecommons.org/licenses/by/4.0/legalcode.txt -O LICENSE
PS C:\repo> wget https://www.toptal.com/developers/gitignore/api/python -O .gitignore
PS C:\repo>
PS C:\repo> 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:\repo> 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
On branch master
nothing to commit, working tree clean
PS C:\repo> git push -u origin master
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), 7.70 KiB | 1.92 MiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:nssedov/matmod.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
PS C:\repo> git push
Everything up-to-date
PS C:\repo>

```

Figure 2.5: Загрузка файлов

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

```

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:/repo/.git/hooks]
PS C:\repo> git branch
* develop
  master
PS C:\repo> 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:\repo> echo "1.0.0" >> version
PS C:\repo>
PS C:\repo> git add .
PS C:\repo> git commit -am "chore(main): add version"
[release/1.0.0 4f36727] chore(main): add version
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 version
PS C:\repo> git flow release finish -m "ver 1" 1.0.0
Switched to branch 'master'

```

Figure 2.6: Инициализация git-flow и начало релиза

```

PS C:\repo> 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), 493 bytes | 246.00 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:nssedov/matmod.git
 * [new branch] master -> master
 * [new branch] develop -> develop
PS C:\repo> git push --tags
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 158 bytes | 158.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:nssedov/matmod.git
 * [new tag] v1.0.0 -> v1.0.0
PS C:\repo>

```

Figure 2.7: Завершение релиза и отправка изменений в сетевой репозиторий

Выполним объединение веток

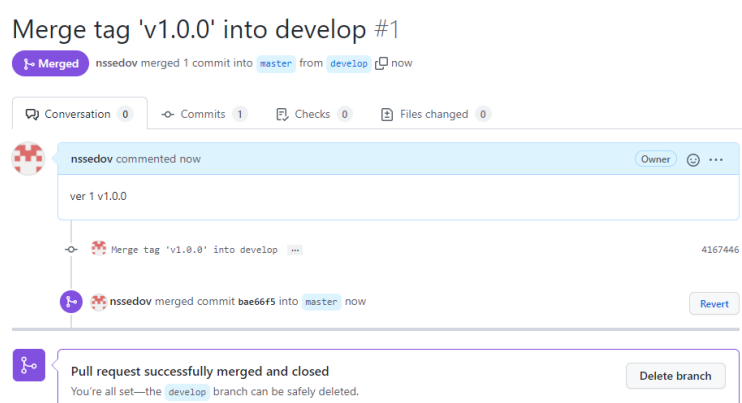


Figure 2.8: Объединение веток в сетевом репозитории



## 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