

# **Отчёт по лабораторной работе №1**

**Система контроля версий Git**

**Чан Куок Кхань НПИбд-02-19**

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>5</b>
<b>3</b>	<b>Вывод</b>	<b>10</b>
	<b>Список литературы</b>	<b>11</b>

# List of Figures

2.1	Создание репозитория . . . . .	5
2.2	Инициализация репозитория . . . . .	6
2.3	Создание SSH-ключа . . . . .	6
2.4	Добавление ключа на github.com . . . . .	7
2.5	Загрузка файлов . . . . .	7
2.6	Инициализация git-flow и начало релиза . . . . .	8
2.7	Завершение релиза и отправка изменений в сетевой репозиторий . . . . .	8
2.8	Объединение веток в сетевом репозитории . . . . .	9

# **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

Create a new repository

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

Owner \* chankuokkhan / Repository name \* matmodel ✓

Great repository names are short, lowercase, and unique. matmodel is available. Need inspiration? How about [friendly-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

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

Figure 2.2: Инициализация репозитория

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

```
PS C:\Users\User\Desktop\work> git init
Initialized empty Git repository in C:/Users/User/Desktop/work/.git/
PS C:\Users\User\Desktop\work> git add README.md
PS C:\Users\User\Desktop\work> git commit -m "first commit"
[master (root-commit) 574280c] first commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 README.md
PS C:\Users\User\Desktop\work> ssh-keygen -C "chankuokkhan 1032189063@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:epjnjG0vU0XATEXPBEf135Iud2wafnDLHbuk3mPluk chankuokkhan 1032189063@pfur.ru
The key's randomart image is:
+--[RSA 2048]-----
  .==+
  .oo+
  o%o.+
  .e.o.
  .+S.
  o+* o
  .++o= +
  BO =.
  oOE.
+--[SHA256]-----
PS C:\Users\User\Desktop\work> cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDAQDF01Tbgbwvng1rcN1zVX3Syod/yw/pn6zGvi2Lpn+wiGk89K6q0MBepfoQ05CbcgmHT9TXZQ+5d+hvL2y
S/p6n8GaInk1qT0023K/5vTA1pT19eqx3bcUqgeaUOMMd0Mni3YtXX8D2b3KyopqMMH8FBd8U3jmac8+Stuxgoa18EKvNB6H2CFYFRP3Chk1Vde+FrCX65L
DhX1213owku+Sr0M8AUvJERwt8a01xp8PvoaSouxy2p2jv8312A5Bb6U2N+H2/RNYgoqxvjdTSQskCUAK25hvaqX2Ry1AHkgzQbYSuo11aqlVs8Ts7Q
19mi1078yLkttUw8m0 chankuokkhan 1032189063@pfur.ru
PS C:\Users\User\Desktop\work>
```

Figure 2.3: Создание SSH-ключа

## SSH keys / Add new

Title

SSS

Key

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDFQjTbgWbdvngirC
NlZVX3SyOd/yw/pn6zGVl2Lpn+WiGk89K6q6MBEpfoq05Cbcg
mHT9TXZQ+5d+hYL2yS/D6n8GAink1qTQO23K/5vTA1pT19eQ
x3bcUBqeaUOMMd6Mini3YtXX8D2bJKyopqWMH8FBd8U3jma
C8+Stuxgoa18EKYnB6H2CFYFRPJCHk1Vde+frCX65LDJhx1215o
WkU+SrOMBmAuVJeFvwt8a0ixp8Pvoa5ouXyJpvjvB3i2A5Bb6U
2N+h2/RNYgoqXvjdTSQsKcUAhC25hvAqX2Ry1AHKgzQbYSu
OliaQLVs8ls7QL9nu1D7RyLhktUWwBnwD chankuokkhan
```

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:\Users\User\Desktop\work> git remote add origin git@github.com:chankuokkhan/matmodel.git
PS C:\Users\User\Desktop\work> wget https://creativecommons.org/licenses/by/4.0/legalcode.txt -O LICENSE
PS C:\Users\User\Desktop\work> wget https://www.toptal.com/developers/gitignore/api/python -O .gitignore
PS C:\Users\User\Desktop\work> 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:\Users\User\Desktop\work> git commit -am "add license"
[master 4477b46] add license
 2 files changed, 555 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 LICENSE
PS C:\Users\User\Desktop\work> git push -u origin master
The authenticity of host 'github.com (140.82.121.4)' can't be established.
ED25519 key fingerprint is SHA256:4b1Y3wvv6TuJJhbpzIsF/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 | 1.54 MiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0
to github.com:chankuokkhan/matmodel.git
 * [new branch] master -> master
branch 'master' set up to track 'origin/master'.
PS C:\Users\User\Desktop\work> git push
Everything up-to-date
PS C:\Users\User\Desktop\work>
```

Figure 2.5: Загрузка файлов

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

```

PS C:\Users\User\Desktop\work> 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:/Users/User/Desktop/work/.git/hooks]
PS C:\Users\User\Desktop\work> git branch
* develop
  master
PS C:\Users\User\Desktop\work> 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\work> echo "1.0.0" >> version
PS C:\Users\User\Desktop\work> git add .
PS C:\Users\User\Desktop\work> git commit -am "chore(main): add version"
[release/1.0.0 7609166] chore(main): add version
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 version
PS C:\Users\User\Desktop\work> 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 1.0.0 >> 16 bytes

```

Figure 2.6: Инициализация git-flow и начало релиза

```

PS C:\Users\User\Desktop\work> 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), 498 bytes | 166.00 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:chankuokkhan/matmodel.git
 44f7b46..fa054d8 master -> master
 * [new branch] develop -> develop
PS C:\Users\User\Desktop\work> git push --tags
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 160 bytes | 160.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:chankuokkhan/matmodel.git
 * [new tag] v1.0.0 -> v1.0.0
PS C:\Users\User\Desktop\work>

```

Figure 2.7: Завершение релиза и отправка изменений в сетевой репозиторий

Выполним объединение веток



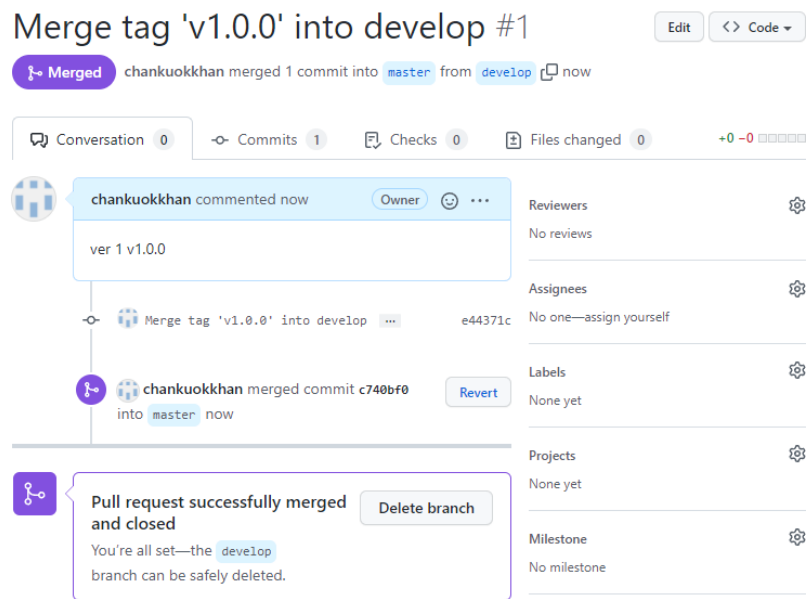


Figure 2.8: Объединение веток в сетевом репозитории

## 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