

Лабораторная работа 1

Искандер Абдуллаев НФИбд-01-19

12 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 * Repository name *

 IskanderAbdullaev ▾ / MatMod ✓

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

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

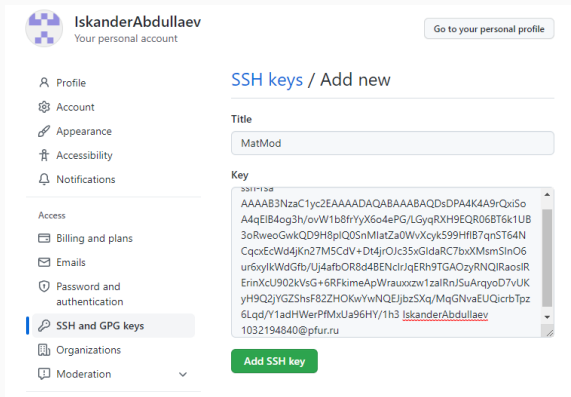
Figure 2: Инициализация репозитория

Создаем SSH-ключ

```
PS C:\Users\User\Documents\MatMod> git init
Initialized empty Git repository in C:\Users\User\Documents\MatMod\.git/
PS C:\Users\User\Documents\MatMod> echo "# Наблюдение паботы" >> README.md
PS C:\Users\User\Documents\MatMod> git add README.md
PS C:\Users\User\Documents\MatMod> git config --global user.name IskanderAbdullaev
PS C:\Users\User\Documents\MatMod> git config --global user.email "1032194840@pfur.ru"
PS C:\Users\User\Documents\MatMod> git commit -m "first commit"
[master (root-commit) 7db3c43] first commit
1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 README.md
PS C:\Users\User\Documents\MatMod> ssh-keygen -C "IskanderAbdullaev 1032194840@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:Ampu3qME1E361pfCHaQzdJVOXOZdu7ZWJ/v6Yo8aE IskanderAbdullaev 1032194840@pfur.ru
The key's randomart image is:
+--[RSA 2048]-----
|
|  .  .  .  .  .  .
| o+  +  +  +  +  +
|o+  o  .  o  .
| o  o  o  +  +
| o  .  S  B
|   O+  o  .
|   +  +  +
|   E  .  .
+-----[SHA256]-----
PS C:\Users\User\Documents\MatMod> cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDDPAK4A9PQX1SoA4qE184og3h/ovw1b8fryyx6o4ePc/LGyqRXH9Eqr068T0k1U83oRweocwKOD9Hsp1
05m1atzZa0Wxcyk39Hf1E1qSTf4Mcqcxcw43kn2W5cdVdrt4jro1c33xG1dabC7bX0xns1n06u6yXjKw6fUj4atBOR846Eic13qEub9fGa
0zyRND1Raos1REfInxCU9D2KVsG6R6K1neApWrauxxw1zaIRn3SuAryqod7vUkyH9Q2jYgZShsF82ZH0KwYwQD1jBzSxq/MqGvvaEUQ1cbrtpZ6Lqd/Y1
adIwEPFmXa96iy/1h3 IskanderAbdullaev 1032194840@pfur.ru
PS C:\Users\User\Documents\MatMod>
```

Figure 3: Создание SSH-ключа

Создаем SSH-ключ



The screenshot shows the GitHub user interface for 'IskanderAbdullaev'. The left sidebar contains navigation links: Profile, Account, Appearance, Accessibility, Notifications, Access, Billing and plans, Emails, Password and authentication, SSH and GPG keys (highlighted), Organizations, and Moderation. The main content area is titled 'SSH keys / Add new'. It features a 'Title' input field with 'MatMod' and a 'Key' input field containing a long SSH key. The key is preceded by 'ssh-rsa' and followed by the email '1032194840@pfur.ru'. A green 'Add SSH key' button is at the bottom right.

IskanderAbdullaev
Your personal account

Go to your personal profile

Profile
Account
Appearance
Accessibility
Notifications

Access
Billing and plans
Emails
Password and authentication
SSH and GPG keys
Organizations
Moderation

SSH keys / Add new

Title
MatMod

Key
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDSDPA4K4A9rQxiSo
A4qElB4og3h/ovW1b8frYyX6o4ePG/LGyqRXH9EQR068T6k1UB
3oRweoGwkQD9H8plQ0SnMlatZa0WvXcyk599HfIB7qnST64N
CqcxEcWd4Jkn27M5CdV+Dt4jrOJc35xGidaRC7bxXMsmSInO6
ur6xylkWdGfb/Uj4afbOR8d4BENclrJqERh9TGAOzyRNQIRaosIR
ErinXcU902kVsG+6RFkimeApWrauxczw1zalRnJSuArqyoD7vUK
yH9Q2YVGZShsF82ZHOKwYwNQEljzbSXq/MqGNvaEUQicrbTpz
6Lqd/Y1adHWerPfMxUa96HY/1h3 IskanderAbdullaev
1032194840@pfur.ru

Add SSH key

Figure 4: Добавление ключа на github.com

Загружаем служебные файлы

```
PS C:\Users\User\Documents\MatMod> git remote add origin git@github.com:IskanderAbdullaev/MatMod.git
PS C:\Users\User\Documents\MatMod> wget https://creativecommons.org/licenses/by/4.0/legalcode.txt -O LICENSE
PS C:\Users\User\Documents\MatMod> wget https://www.topical.com/developers/gitignore/api/python -O .gitignore
PS C:\Users\User\Documents\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:\Users\User\Documents\MatMod> git commit -am "add license"
[master 383ed9b] add license
 2 files changed, 555 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 LICENSE
PS C:\Users\User\Documents\MatMod> git push -u origin master
The authenticity of host 'github.com (140.82.121.3)' can't be established.
ED25519 key fingerprint is SHA256:4d13evvv6tUzJhbpzISF/zLDA02PMSvHdkr4UvCOQu.
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: 2, 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), 7.72 KiB | 1.93 MiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com: IskanderAbdullaev/MatMod.git
 * [new branch] master -> master
branch 'master' set up to track 'origin/master'.
PS C:\Users\User\Documents\MatMod> git push
Everything up-to-date
PS C:\Users\User\Documents\MatMod>
```

Figure 5: Загрузка файлов лицензии и gitignore

Использование системы управления версиями

```
PS C:\Users\User\Documents\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:/Users/User/Documents/MatMod/.git/hooks]
PS C:\Users\User\Documents\MatMod> git branch
* develop
  master
PS C:\Users\User\Documents\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:\Users\User\Documents\MatMod> echo "1.0.0" >> version
PS C:\Users\User\Documents\MatMod> git add .
PS C:\Users\User\Documents\MatMod> git commit -am "main: add version"
[release/1.0.0 ddd9alb] main: add version
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 version
PS C:\Users\User\Documents\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.
(use "git push" to publish your local commits)
```

Figure 6: Инициализация git-flow и создание релиза

Использование системы управления версиями

```
PS C:\Users\User\Documents\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), 476 bytes | 158.00 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:IskanderAbdullaev/MatMod.git
 * 383ed8b..3649317 master -> master
 * [new branch]      develop -> develop
PS C:\Users\User\Documents\MatMod> git push --tags
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 167 bytes | 167.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:IskanderAbdullaev/MatMod.git
 * [new tag]         v1.0.0 -> v1.0.0
PS C:\Users\User\Documents\MatMod>
```

Figure 7: Отправка изменений в сетевой репозиторий

Выполним объединение веток

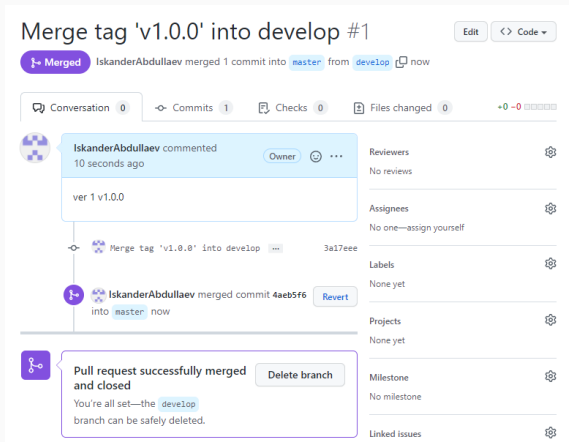


Figure 8: Объединение веток в сетевом репозитории

Выводы по проделанной работе

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