Лабораторная работа №1

Маммедгулыев Максат НФИбд-02-19

12 февраля, 2022, Москва, Россия

Российский Университет Дружбы Народов

Цели и задачи работы

Цель лабораторной работы

Целью данной лабораторной работы является изучение пространства git, применение раздичных команд и отработка эллементарных действий с системой.

Задачи лабораторной работы

1. Создать учетную запись на github и репозиторий
2. Настроить репозиторий и создать файл для работы дальнейшей
3. Изучить механизм управления версиями и применить простые комманды

Процесс выполнения лабораторной работы

Создаем учетную запись на github.com и репозиторий, в котором в дальнейшем будем работать

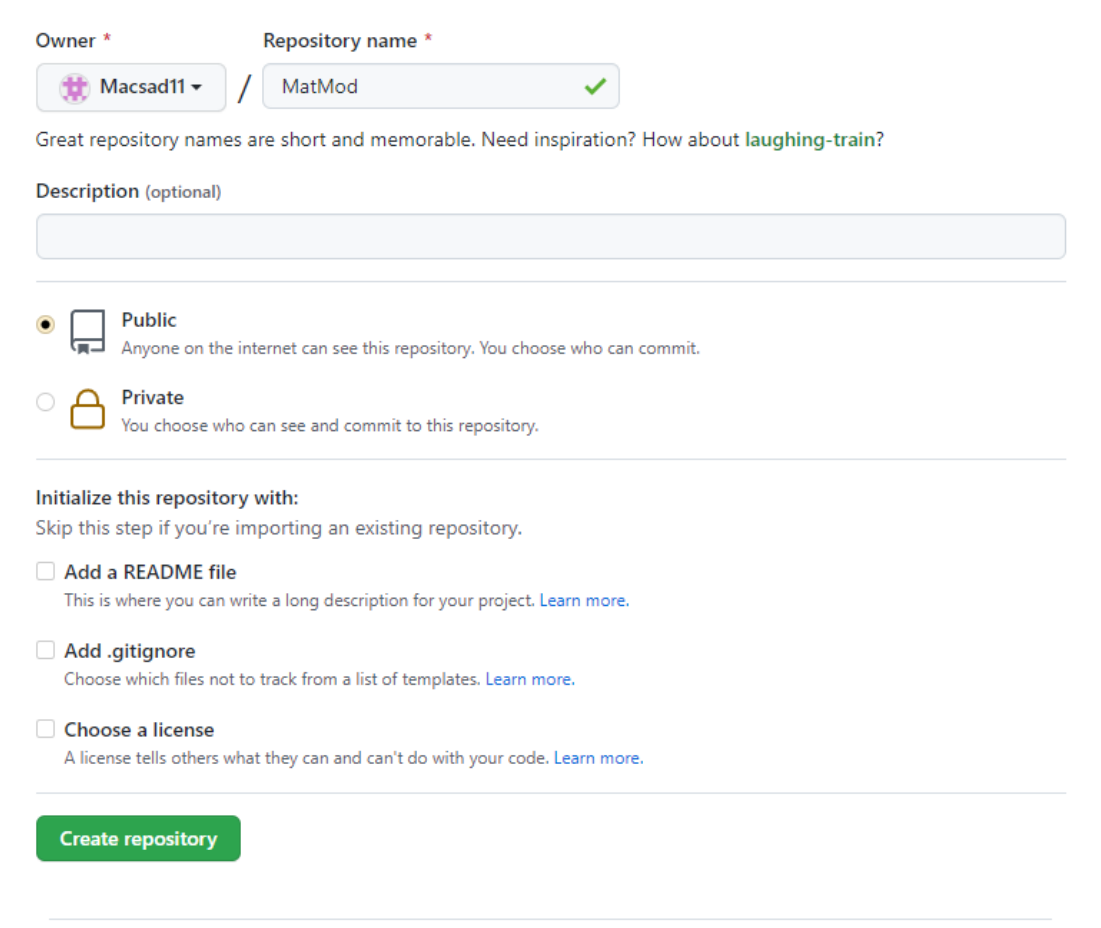

Figure 1: Создание учетной записи и репозитория

Инициализируем созданный нами локальный репозиторий, после создаем в нем файл README.md, с которым будем проводить все дальнейшие действия

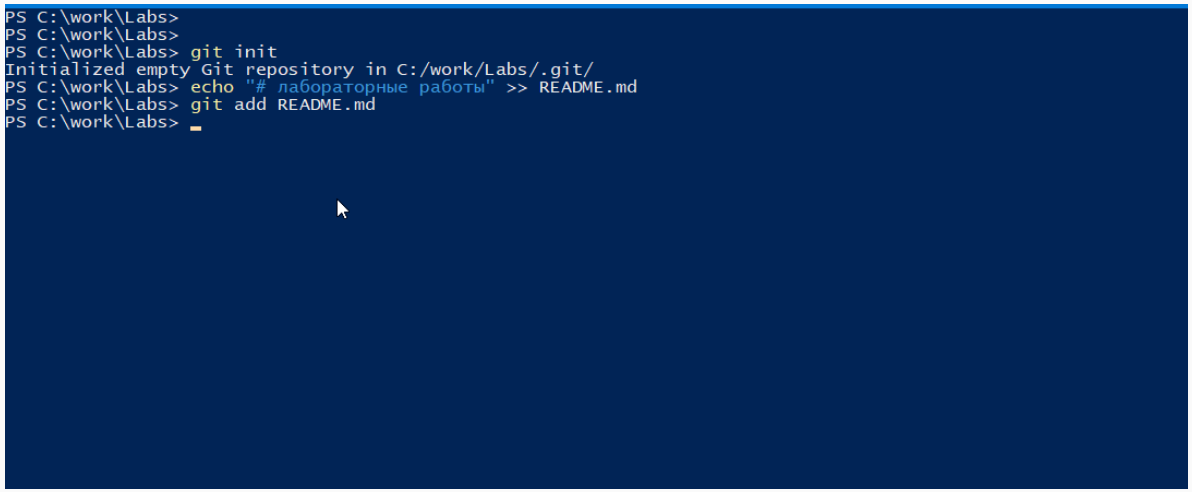

Figure 2: Инициализация репозитория и создание файла

Создаем SSH-ключ, который пропишем в настройках на github.com

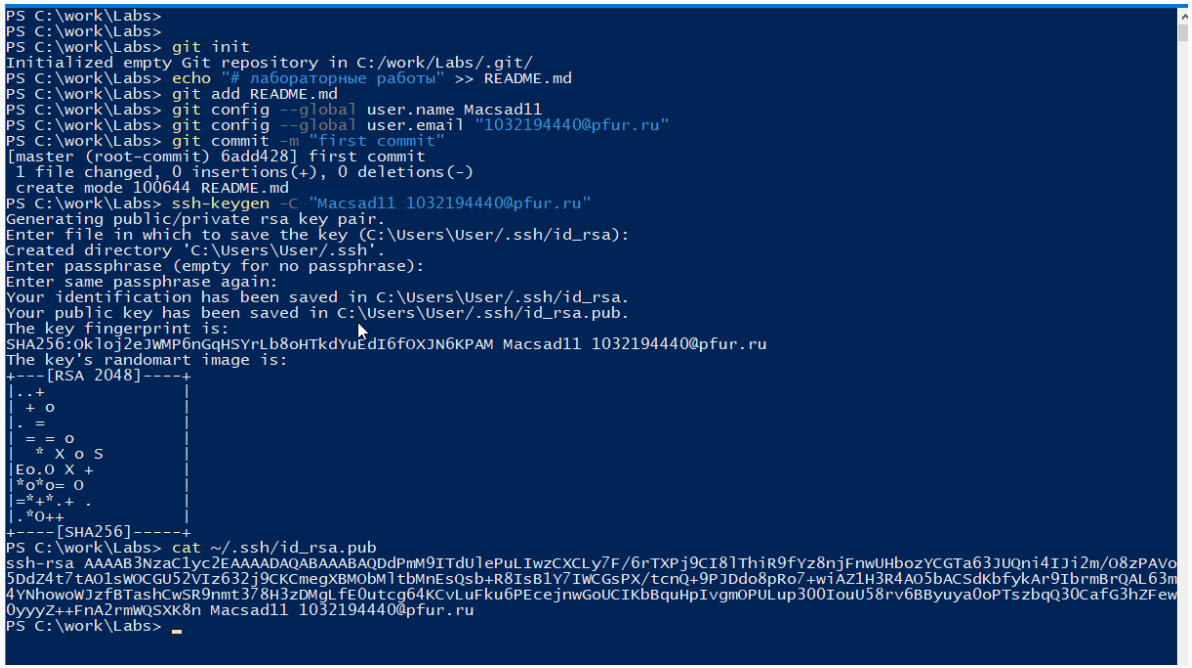

Figure 3: Создание SSH-ключа

Создание ключа

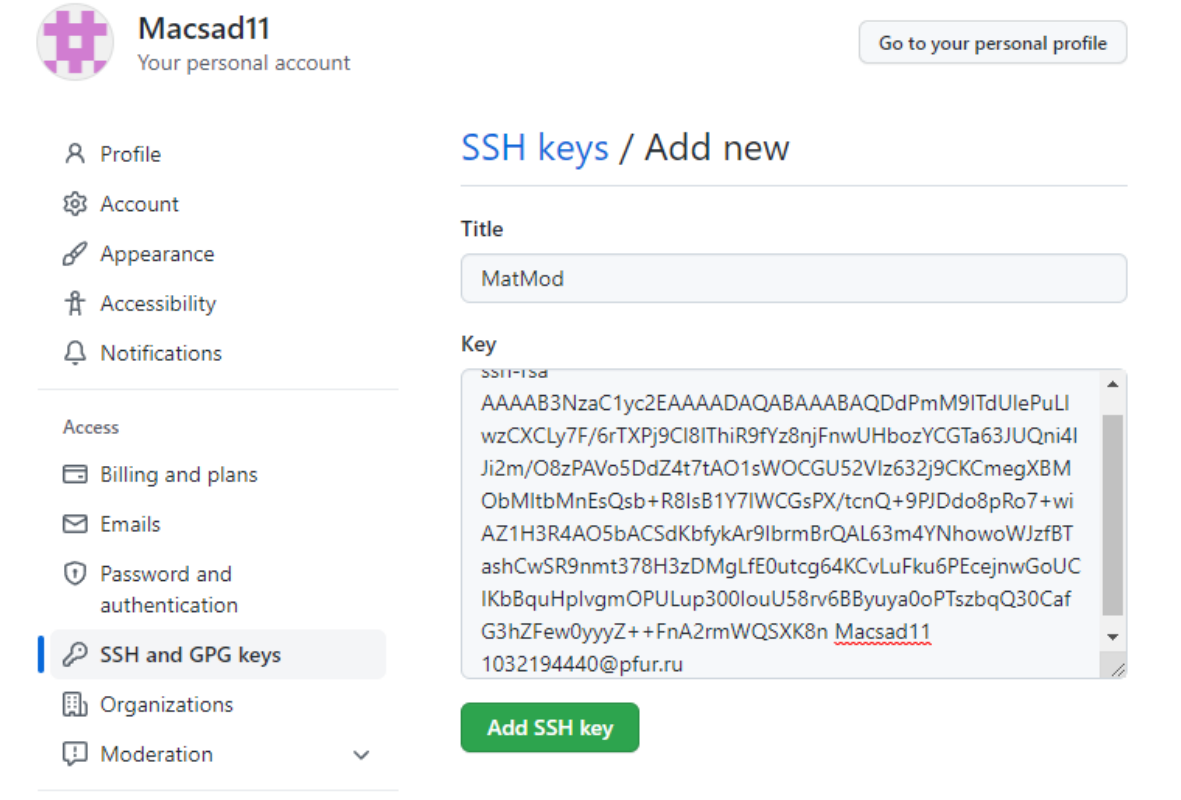

Figure 4: Добавляем ключ на сайт

Загружаем файлы лицензионного соглашения и gitignore и отправляем эти файлы в сетевой репозиторий.

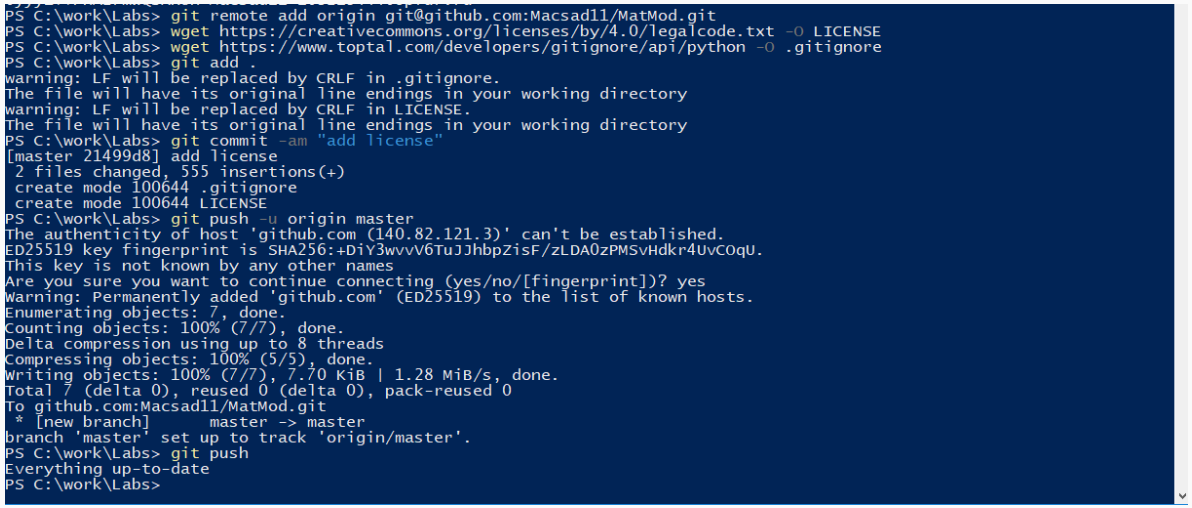

Figure 5: Загрузка файлов с дальнейшей отправкой в сетевой репозиторий

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

```
PS C:\work\Labs> 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:/work/Labs/.git/hooks]
PS C:\work\Labs> git branch
* develop
  master
PS C:\work\Labs> 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:\work\Labs> echo "1.0.0" >> version
PS C:\work\Labs> git add .
PS C:\work\Labs> git commit -am "main: add version"
[release/1.0.0 60c3196] main: add version
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 version
PS C:\work\Labs> 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.
```

Figure 6: Инициализируем git-flow и создание релиза

Используем систему управления версиями

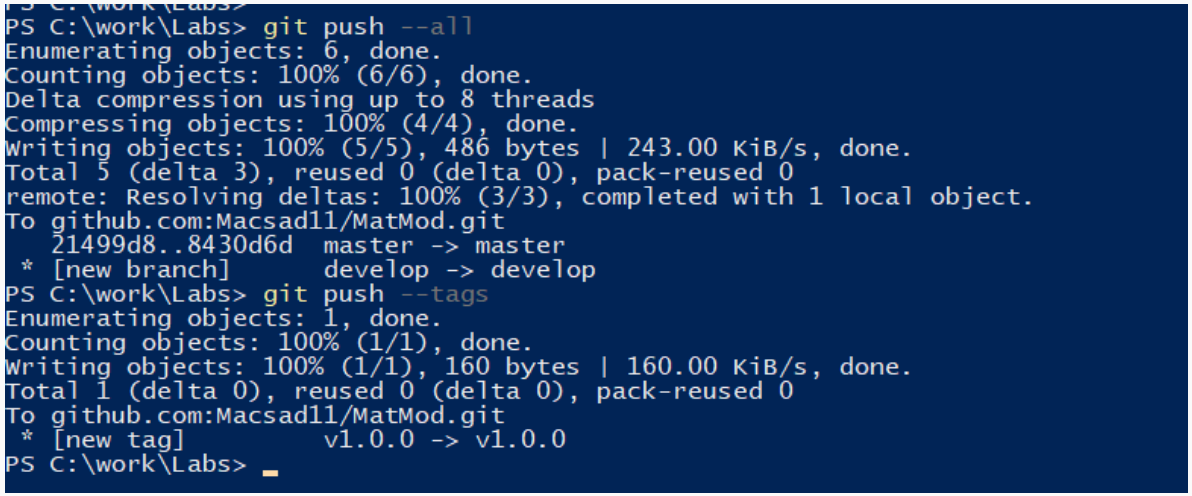

Figure 7: Отправляем изменения в сетевой репозиторий

Выполняем объединение веток

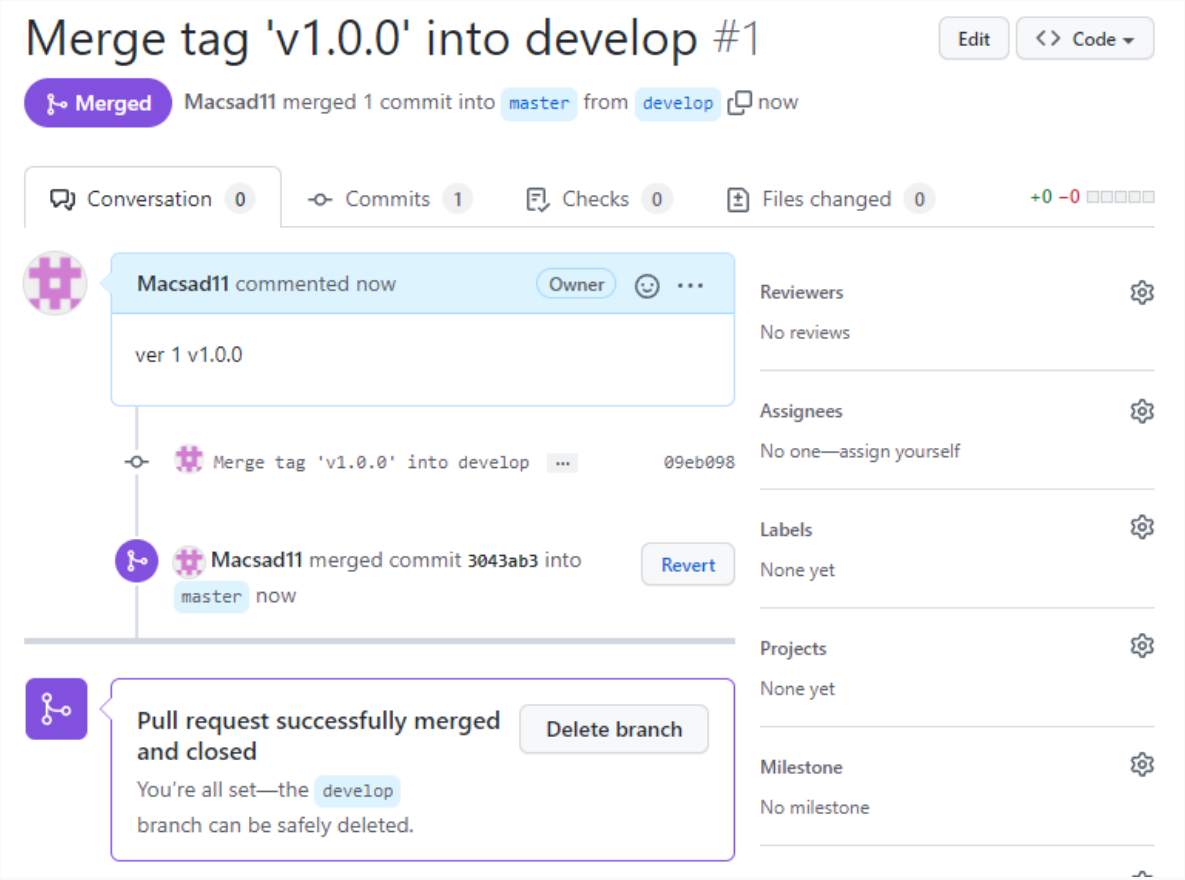

Figure 8: Объединяем ветки в сетевом репозитории

Сделаем выводы по проделанной работе в лабораторной №1:

Мы приобрели практические навыки работы с системой контроля версий git и создали свой репозиторий. Изучили пространства git, применили различные команды и отработали эллементарные действий с системой.