

Отчёт по лабораторной работе №2

Серегин Денис Алексеевич

Содержание

1	Цель работы	5
2	Задания	6
3	Выполнение лабораторной работы	7
4	Выводы	15
5	Использованные ресурсы	16

List of Tables

List of Figures

3.1	Создание репозитория на github	7
3.2	Переход в рабочий каталог	8
3.3	Инициализация системы git	8
3.4	Создание заготовки для файла README	8
3.5	Создание заготовки для файла README	8
3.6	Создание коммита	9
3.7	Добавление файла лицензии	9
3.8	Добавление шаблона gitignore	10
3.9	Добавление шаблона для C	10
3.10	Создание заготовки для файла README	10
3.11	Отправка на github	11
3.12	Инициализация git-flow	11
3.13	Проверка ветки	11
3.14	Создание релиза с версией 1.0.0	12
3.15	Запись версии	12
3.16	Добавление файлов	12
3.17	Добавление коммита	12
3.18	Залив релизной ветки в основную	13
3.19	Отправка на github	13
3.20	Релиз	14

1 Цель работы

Изучить идеологию и применение средств контроля версий.

2 Задания

1. Настроить git.
2. Подключить репозиторий к github.
3. Провести первичную конфигурацию.
4. Провести конфигурацию git-flow.

3 Выполнение лабораторной работы

1. Создал учётную запись на github.
2. Настроил систему контроля версий так как это описано в указаниях к лабораторной работе.
3. Создал структуру каталога лабораторных работ согласно пункту М.2.
4. Создал репозиторий на GitHub. Для примера назвал его os-intro.

(рис. 3.1)

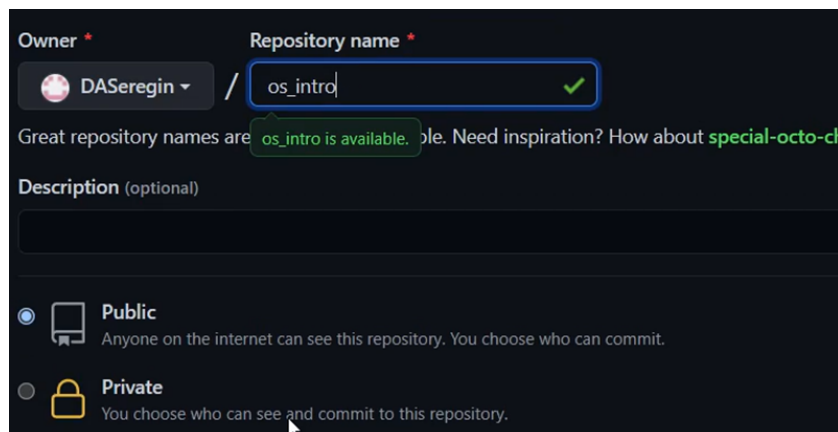


Figure 3.1: Создание репозитория на github

5. Рабочий каталог обозначил как laboratory и перешел в него

(рис. 3.2)

```
C:\Windows\system32>d:

D:\>cd РУДН

D:\РУДН>cd os

D:\РУДН\OS>cd labratory
```

Figure 3.2: Переход в рабочий каталог

6. Инициализировал систему git

(рис. 3.3)

```
D:\РУДН\OS\labratory>git init
Initialized empty Git repository in D:/РУДН/OS/labratory/.git/
```

Figure 3.3: Инициализация системы git

7. Создал заготовку для файла README.md

(рис. 3.4)

```
D:\РУДН\OS\labratory>echo "# Лабораторные работы" >> README.md
```

Figure 3.4: Создание заготовки для файла README

(рис. 3.5)

```
D:\РУДН\OS\labratory>git add README.md
```

Figure 3.5: Создание заготовки для файла README

8. Сделал первый коммит и выложил на github

(рис. 3.6)


```

D:\ПУДН\OS\laboratory>git commit -m "first commit"
[master (root-commit) 946b371] first commit
1 file changed, 1 insertion(+)
create mode 100644 README.md

D:\ПУДН\OS\laboratory>git remote add origin https://github.com/DASeregin/os_intro.git

D:\ПУДН\OS\laboratory>git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 244 bytes | 244.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/DASeregin/os_intro.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.

```

Figure 3.6: Создание коммита

9. Добавил файл лицензии

(рис. 3.7)

```

D:\ПУДН\OS\laboratory>wget https://creativecommons.org/licenses/by/4.0/legalcode.txt -O LICENSE
--2021-06-06 18:02:19-- https://creativecommons.org/licenses/by/4.0/legalcode.txt
Resolving creativecommons.org (creativecommons.org)... 104.20.151.16, 172.67.34.140, 104.20.150
.16
Connecting to creativecommons.org (creativecommons.org)|104.20.151.16|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/plain]
Saving to: 'LICENSE'

LICENSE                               [ <=> ] 18,22K 55,3KB/s in 0,3s

```

Figure 3.7: Добавление файла лицензии

10. Добавил шаблон игнорируемых файлов. Просмотрел список имеющихся шаблонов

(рис. 3.8)

```
D:\ПУДН\OS\labratory>curl -L -s https://www.gitignore.io/api/list
1c,1c-bitrix,a-frame,actionsript,ada
adobe,advancedinstaller,adventuregamestudio,agda,al
alteraquartusii,altium,amplify,android,androidstudio
angular,anjuta,ansible,ansibletower,apachecordova
apachehadoop,appbuilder,appcelerator titanium,appcode,appcode+all
appcode+iml,appengine,aptanastudio,arcanist,archive
archives,archlinuxpackages,aspnetcore,assembler,ate
atmelstudio,ats,audio,automationstudio,autotools
autotools+strict,awr,azurefunctions,backup,ballerina
basercms,basic,batch,bazaar,bazel
bitrise,bitrix,bittorrent,blackbox,bloop
bluej,bookdown,bower,bricxcc,buck
c,c++,cake,cakephp,cakephp2
cakephp3,calabash,carthage,certificates,ceylon
cfwheels,chefcookbook,chocolatey,clean,clion
clion+all,clion+iml,clojure,cloud9,cmake
cocoapods,cocos2dx,cocoscreator,code-java,codeblocks
```

Figure 3.8: Добавление шаблона gitignore

11. Затем скачал шаблон, например, для C и добавил новые файлы
(рис. 3.9)

```
D:\ПУДН\OS\labratory>curl -L -s https://www.gitignore.io/api/c >> .gitignore
D:\ПУДН\OS\labratory>git add .
```

Figure 3.9: Добавление шаблона для C

12. Выполнил коммит
(рис. 3.10)

```
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# Your branch is up to date with 'origin/master'.
#
# Changes to be committed:
#   new file:   .gitignore
#   new file:   LICENSE
#
lab_
```

Figure 3.10: Создание заготовки для файла README

13. Отправил на github
(рис. 3.11)

```
D:\ПУДН\OS\labratory>git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 6.43 KiB | 6.43 MiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/DASeregin/os_intro.git
946b371..aeac79c master -> master
```

Figure 3.11: Отправка на github

14. Инициализировал git-flow. Префикс для ярлыков установил в v.

(рис. 3.12)

```
D:\ПУДН\OS\labratory>git flow init

Which branch should be used for bringing forth production releases?
- master
Branch name for production releases: [master] develop
Fatal: Local branch 'develop' does not exist.

D:\ПУДН\OS\labratory>git flow init

Which branch should be used for bringing forth production releases?
- master
Branch name for production releases: [master] develop:
Fatal: Local branch 'develop:' does not exist.

D:\ПУДН\OS\labratory>git flow init

Which branch should be used for bringing forth production releases?
- master
```

Figure 3.12: Инициализация git-flow

15. Проверил, что нахожусь на ветке develop

(рис. 3.13)

```
D:\ПУДН\OS\labratory>git branch
* develop
master
```

Figure 3.13: Проверка ветки

16. Создал релиз с версией 1.0.0

(рис. 3.14)

```
D:\ПУДН\OS\labratory>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'
```

Figure 3.14: Создание релиза с версией 1.0.0

17. Записал версию

(рис. 3.15)

```
D:\ПУДН\OS\labratory>echo "1.0.0" >> VERSION
```

Figure 3.15: Запись версии

18. Добавил в индекс

(рис. 3.16)

```
D:\ПУДН\OS\labratory>git add .
warning: CRLF will be replaced by LF in VERSION.
The file will have its original line endings in your working directory
```

Figure 3.16: Добавление файлов

(рис. 3.17)

```
D:\ПУДН\OS\labratory>git commit -am "chore(main): add version"
[release/1.0.0 401af3d] chore(main): add version
1 file changed, 1 insertion(+)
create mode 100644 VERSION
```

Figure 3.17: Добавление коммита

19. Залил релизную ветку в основную ветку

(рис. 3.18)

```
D:\ПУДН\OS\labratory>git flow release finish 1.0.0
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
Merge made by the 'recursive' strategy.
 LICENSE => LICENSE.txt | 1 -
  VERSION                | 1 +
 2 files changed, 1 insertion(+), 1 deletion(-)
 rename LICENSE => LICENSE.txt (99%)
 create mode 100644 VERSION
```

Figure 3.18: Залив релизной ветки в основную

20. Отправил данные на github

(рис. 3.19)

```
D:\ПУДН\OS\labratory>git push --all
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done. [+] [unix] (19:06 06/06/2021)
Delta compression using up to 8 threads [unix] 6L, 86B written
Compressing objects: 100% (12/12), done.
Writing objects: 100% (15/15), 7.40 KiB | 2.47 MiB/s, done.
Total 15 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), done.
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:   https://github.com/DASeregin/os_intro/pull/new/develop
remote:
To https://github.com/DASeregin/os_intro.git
 * [new branch]      develop -> develop
 ! [rejected]        master -> master (fetch first)
```

Figure 3.19: Отправка на github

21. Создал релиз на github.

(рис. 3.20)



Figure 3.20: Релиз

4 Выводы

Благодаря лабораторной работе я смог изучить идеологию и применить систему контроля версий git.

5 Используемые ресурсы

<https://habr.com/ru/post/125799/>