



МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технологический университет «СТАНКИН»
(ФГБОУ ВО «МГТУ «СТАНКИН»)

Институт цифровых
интеллектуальных систем

Кафедра
компьютерных систем управления

Дисциплина «Основы системного программного обеспечения»

Отчет по лабораторной работе №1

На тему: «Работа с системами контроля версий на примере Git Hub»

Направление: 15.03.04 «Автоматизация технологических процессов и
производств»

Выполнил
студент гр. АДБ-20-07:

(дата)

(подпись)

Громов Д.А.

Проверил
к.т.н., доцент

(дата)

(подпись)

Ковалев И.А.

Москва 2023 г.

Оглавление

Цель работы:	3
Основные задачи работы:	3
Ход работы:	4
Вывод:	12

Цель работы:

работа с системами контроля версий на примере Git Hub

Основные задачи работы:

1. Создать учетную запись на github.com
2. Создать локальный репозиторий
3. Зафиксировать изменения в области заготовленных файлов
4. Переслать локальный коммит на сервер
5. Сделать слияние веток
6. Просмотреть изменения и разрешение конфликтов
7. Удалить ветку на сервере и вернуть к предыдущему состоянию
8. Исправить коммит

Ход работы:

1. Создаем учетную запись на github.com или заходим в уже имеющийся личный кабинет, как в нашем случае и создаем новый репозиторий.
2. Создали файл test.txt и написали наши фамилии. Открыли командную строку «cmd» и перешли с помощью команды «cd» в созданный нами каталог. Настроим самые важные опции и параметры: наше имя пользователя и адрес электронной почты.
3. Переходим в созданную нами папку, используя команду cd. Теперь необходимо инициализировать эту папку как git репозиторий. Вводим команду: git init. Командная строка возвращает нам сообщение, что инициализирована пустая git директория. Теперь используя проводник windows создаем в своей папке текстовый файл и пишем в него наши имена. Вводим команду: git status и нам отображается сообщение, что есть новый файл, но он не отслеживается. Чтобы исправить ситуацию используем команды add . и commit.

```

Microsoft Windows [Version 10.0.19045.2965]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\Михаил>C:\lapka
"C:\lapka" не является внутренней или внешней
командой, исполняемой программой или пакетным файлом.

C:\Users\Михаил>cd C:\lapka

C:\lapka>git init
Initialized empty Git repository in C:/lapka/.git/

C:\lapka>git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        1.txt

nothing added to commit but untracked files present (use "git add" to track)

C:\lapka>git add .

C:\lapka>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   1.txt

C:\lapka>git commit -m "first commit"
[master (root-commit) 9031591] first commit
 1 file changed, 1 insertion(+)
 create mode 100644 1.txt

C:\lapka>git status
On branch master
nothing to commit, working tree clean

C:\lapka>git remote add origin https://github.com/Shpyn/automatic-octo-carnival.git

C:\lapka>git remote -v
origin  https://github.com/Shpyn/automatic-octo-carnival.git (fetch)
origin  https://github.com/Shpyn/automatic-octo-carnival.git (push)

```

4. Чтобы связать наш локальный репозиторий с репозиторием на GitHub, выполним следующую команду в терминале.

```
C:\lapka>git remote add origin https://github.com/Shpyn/automatic-octo-carnival.git
```

5. Запросим изменения с сервера

Мы создали изменения в нашем репозитории, создав файл readme и другие пользователи могут скачать изменения при помощи команды pull.

```

C:\lapka>git pull origin master --allow-unrelated-histories
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 606 bytes | 86.00 KiB/s, done.
From https://github.com/Shpyn/automatic-octo-carnival
 * branch          master      -> FETCH_HEAD
 * [new branch]    master      -> origin/master
Merge made by the 'ort' strategy.
 README.md | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

```

6. Посмотрим наши изменения

```

C:\lapka>git log
commit e1fdca26fa78bfc2fec8f08b43100b513a29d088 (HEAD -> master, origin/master)
Merge: 9031591 abc5763
Author: Mikhail <mxailkorneevga@gmail.com>
Date:   Sat Jun 3 17:56:39 2023 +0300

    Merge branch 'master' of https://github.com/Shpyn/automatic-octo-carnival

commit 90315912b8c9969c26b4b36986d4bfe643ed7cc9
Author: Mikhail <mxailkorneevga@gmail.com>
Date:   Sat Jun 3 17:55:11 2023 +0300

    first commit

commit abc5763cc79182a1fcee4adc48fdb9471d11b80f
Author: Shpyn <92688003+Shpyn@users.noreply.github.com>
Date:   Sat Jun 3 17:53:57 2023 +0300

    Initial commit

```

7. Создание новой ветки

Создадим новую ветку second, посмотрим в какой находимся и перейдем на другую:

```

C:\lapka>git branch second

C:\lapka>git branch
* master
  second

C:\lapka>git checkout second
Switched to branch 'second'

```

8. Создаем новый файл в нашем локальном репозитории и пишем в нем свои фамилии, добавляем в область подготовленных файлов, коммитим и

отправляем на сервер:

так же сливаем ветки и удаляем second

```
C:\lapka>git add .

C:\lapka>git commit -m "second commit"
[second 7c69e50] second commit
1 file changed, 1 insertion(+)
create mode 100644 2.txt

C:\lapka>git push origin second
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 320 bytes | 320.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'second' on GitHub by visiting:
remote:   https://github.com/Shpyn/automatic-octo-carnival/pull/new/second
remote:
To https://github.com/Shpyn/automatic-octo-carnival.git
 * [new branch]      second -> second

C:\lapka>git branch master
fatal: a branch named 'master' already exists

C:\lapka>git checkout master
Switched to branch 'master'

C:\lapka>git merge second
Updating e1fdca2..7c69e50
Fast-forward
 2.txt | 1 +
1 file changed, 1 insertion(+)
create mode 100644 2.txt

C:\lapka>git branch -d second
Deleted branch second (was 7c69e50).
```

9. Теперь представив, что задание усложняется, предполагаем, что в двух ветках могут быть одинаковые файлы и над ними работают разные разработчики.

Создаем ветку с названием newdev. Переключаемся на нее. Добавляем в файл с именами и отчествами. Фиксируем и коммитим изменения.

```

C:\lapka>git branch newdev

C:\lapka>git checkout newdev
Switched to branch 'newdev'

C:\lapka>git add .

C:\lapka>git commit -m "izm"
[newdev 46ad206] izm
 1 file changed, 1 insertion(+), 1 deletion(-)

C:\lapka>git checkout master
Switched to branch 'master'

C:\lapka>git merge newdev
Updating 7c69e50..46ad206
Fast-forward
 1.txt | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)

C:\lapka>git checkout newdev
Switched to branch 'newdev'

C:\lapka>git add .

C:\lapka>git commit -m "izm1"
[newdev 638e419] izm1
 1 file changed, 1 insertion(+), 1 deletion(-)

C:\lapka>git checkout master
Switched to branch 'master'

C:\lapka>git add .

C:\lapka>git commit -m "izm2"
[master b7fb63b] izm2
 1 file changed, 1 insertion(+), 1 deletion(-)

```

10. Просмотр изменений и разрешение конфликтов

Объединим ветки `git merge newdev`. Теперь ничего не получится, т.к. есть изменения в обеих ветках. Нужно разрешить конфликты:

```

C:\lapka>git merge newdev
Auto-merging 1.txt
CONFLICT (content): Merge conflict in 1.txt
Automatic merge failed; fix conflicts and then commit the result.

```

Наберем команду для просмотра изменений `git dif`:


```

C:\lapka>git diff
diff --cc 1.txt
index 1d1a9e9,f36e939..0000000
--- a/1.txt
+++ b/1.txt
@@@ -1,1 -1,1 +1,5 @@@
- Mikhail Andreevich123
- Mikhail Andreev
++<<<<<<< HEAD
++Mikhail Andreevich123
++=====
++Mikhail Andreev
++>>>>>>> newdev

```

Приложение отметило строки, содержащие конфликт:

Над разделителем ===== мы видим последний (HEAD) коммит, а под ним — конфликтующий. Перепишем все, удалив разделители (HEAD, ==, <>), и дадим git понять, что закончили.

Удаляем ветку newdev:

```

C:\lab1.ospo>git branch -d newdev
Deleted branch newdev (was 254a88f).

```

11. Удаление веток на сервере

Удаляем ветки с github:

```

C:\lapka>git push origin --delete newdev
error: unable to delete 'newdev': remote ref does not exist
error: failed to push some refs to 'https://github.com/Shpyn/automatic-octo-carnival.git'

C:\lapka>git push origin --delete second
To https://github.com/Shpyn/automatic-octo-carnival.git
- [deleted]          second

```

12. Возврат к предыдущему состоянию

Чтобы посмотреть все комиты, можно использовать команду git log:

```

C:\lapka>git log
commit 0781c7492b9db681ce1b86d41a68aa02ef77bf04 (HEAD -> master, origin/master)
Merge: b7fb63b 638e419
Author: Mikhail <mxailkorneevga@gmail.com>
Date: Sat Jun 3 18:20:49 2023 +0300

    izm3

commit b7fb63b6fa8d704df02e4d7d135aeeb5cda43cd6
Author: Mikhail <mxailkorneevga@gmail.com>
Date: Sat Jun 3 18:13:51 2023 +0300

    izm2

commit 638e419a2e90c5843f1d1cc8107248c29941b17d (newdev)
Author: Mikhail <mxailkorneevga@gmail.com>
Date: Sat Jun 3 18:13:22 2023 +0300

    izm1

commit 46ad20640e95b3302a206e067e9042adc5b432ef
Author: Mikhail <mxailkorneevga@gmail.com>
Date: Sat Jun 3 18:10:49 2023 +0300

    izm

commit 7c69e5074a3c1180ddd5531cd5d4bc15940e3b46
Author: Mikhail <mxailkorneevga@gmail.com>
Date: Sat Jun 3 17:59:07 2023 +0300

    second commit

commit e1fdca26fa78bfc2fec8f08b43100b513a29d088
Merge: 9031591 abc5763
Author: Mikhail <mxailkorneevga@gmail.com>
Date: Sat Jun 3 17:56:39 2023 +0300

    Merge branch 'master' of https://github.com/Shpyn/automatic-octo-carnival

commit 90315912b8c9969c26b4b36986d4bfe643ed7cc9
Author: Mikhail <mxailkorneevga@gmail.com>
Date: Sat Jun 3 17:55:11 2023 +0300

    first commit

commit abc5763cc79182a1fcee4adc48fdb9471d11b80f
Author: Shpyn <92688003+Shpyn@users.noreply.github.com>
Date: Sat Jun 3 17:53:57 2023 +0300

    Initial commit

```

Выбираем коммит и откатываемся:

```

C:\lapka>git checkout b7fb63befa8d704df02e4d7d135aeeb5cda43cd6
Note: switching to 'b7fb63befa8d704df02e4d7d135aeeb5cda43cd6'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at b7fb63b izm2

C:\lapka>git branch
* (HEAD detached at b7fb63b)
  master
  newdev

```

13. Отправка только нужных файлов на сервер

Создаем вручную файл под названием «.gitignore» и сохраняем его в директорию проекта. Внутри файла перечисляем названия файлов/папок, которые нужно игнорировать, каждый с новой строки. Файл «.gitignore» добавляем, коммитим и отправляем на сервер.

```

C:\lapka>git commit -m "ignoreme"
[detached HEAD 6fd98e5] ignoreme
1 file changed, 1 insertion(+)
create mode 100644 .gitignore

C:\lapka>git push origin master
Everything up-to-date

```

Вывод: мы провели работу с системами контроля версий на примере Git Hub и научились создавать локальный репозиторий, пересылать локальный коммит на сервер и делать слияние веток.