

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

**«Исследование возможностей Git для работы с локальными
репозиториями»**

ОТЧЕТ
по лабораторной работе №1_3
дисциплины
«Основы программной инженерии»

Выполнил:

Разворотников Денис Сергеевич

2 курс, группа ПИЖ-б-о-21-1,

09.03.04 «Программная инженерия»,

направленность (профиль) «Разработка

и сопровождение программного

обеспечения», очная форма обучения

(подпись)

Проверил:

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Цель работы: исследование базовых возможностей по работе с локальными и удаленными ветками Git.

Ход работы:

1. Проиндексировать первый файл и сделать коммит с комментарием "add 1.txt file".

```
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git add 1.txt
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git status
On branch main
Your branch is up to date with 'origin/main'.

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

[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git commit -m "add 1.txt file"
[main fee9fdc] add 1.txt file
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 1.txt
denisrazvorotnikov@MacBook-Air-Denis lab1_3 %
```

Рисунок 1 – Индексация первого файла и создание коммита

2. Проиндексировать второй и третий файлы.

```
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git add 2.txt
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git add 3.txt
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   2.txt
    new file:   3.txt

denisrazvorotnikov@MacBook-Air-Denis lab1_3 %
```

Рисунок 2 – Добавление файлов 2 и 3 в индекс

3. Переписать уже сделанный коммит с новым комментарием "add 2.txt and 3.txt."

```
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git commit --amend -m "add 2.txt and 3.txt"]
[main 418bb43] add 2.txt and 3.txt
Date: Sat Nov 12 09:58:11 2022 +0300
3 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 1.txt
create mode 100644 2.txt
create mode 100644 3.txt
```

Рисунок 3 – Перезапись коммита

4. Создать новую ветку my_first_branch.

```
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git branch my_first_branch]
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git checkout my_first_branch]
Switched to branch 'my_first_branch'
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % ]
```

Рисунок 4 – Создание новой ветки

5. Перейти на ветку и создать новый файл in_branch.txt, закоммитить изменения.

```
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git status]
On branch my_first_branch
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    git
    in_branch.txt

nothing added to commit but untracked files present (use "git add" to track)
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git add in_branch.txt]
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git commit -m "Add file in_branch.txt"]
[my_first_branch 84622b8] Add file in_branch.txt
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 in_branch.txt
```

Рисунок 5 – Создание коммита в новой ветке

6. Вернуться на ветку master.

```
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git checkout main]
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % ]
```

Рисунок 6 – Возвращение на ветку «main»

7. Создать и сразу перейти на ветку new_branch.


```
C:\Users\ynakh\OneDrive\Рабочий стол\Git\Lab-3-OPJ>git checkout -b new_branch
Switched to a new branch 'new_branch'

C:\Users\ynakh\OneDrive\Рабочий стол\Git\Lab-3-OPJ>git branch
  main
  my_first_branch
* new_branch

C:\Users\ynakh\OneDrive\Рабочий стол\Git\Lab-3-OPJ>
```

Рисунок 7 – Одновременное создание и переход на ветку

8. Сделать изменения в файле 1.txt, добавить строку “new row in the 1.txt file”, закоммитить изменения.



The image shows a screenshot of a text editor window. The title bar at the top right indicates the file is named '1.txt'. The main text area of the editor contains the single line of text: 'new row in the 1.txt file'.

Рисунок 8 – Создание изменений в файле 1.txt и коммита

9. Удалить ветки my_first_branch и new_branch.

```
denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git branch -d my_first_branch
Deleted branch my_first_branch (was 84622b8).
denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git branch -d new_branch
Deleted branch new_branch (was 22646d8).
```

Рисунок 9 – Удаление веток

10. Создать ветки branch_1 и branch_2.

```

[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git branch branch_1
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git branch branch_2
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git checkout branch_1
Switched to branch 'branch_1'
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git status
On branch branch_1
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   1.txt
        modified:   3.txt

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

no changes added to commit (use "git add" and/or "git commit -a")
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git add 1.txt
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git add 3.txt
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git commit -m " Fix 1.txt and 3.txt"
[branch_1 0891842] Fix 1.txt and 3.txt
 2 files changed, 2 insertions(+), 1 deletion(-)
[denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git checkout branch_2
Switched to branch 'branch_2'

```

Рисунок 10 – Создание новых веток и слияние

11. Отправить ветку branch_1 на GitHub.

```

denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git push git@github.com:DenisRazvorotnikov/lab1_3.git
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done.
Delta compression using up to 8 threads
Compressing objects: 100% (10/10), done.
Writing objects: 100% (14/14), 1.20 KiB | 1.20 MiB/s, done.
Total 14 (delta 5), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (5/5), done.
remote:
remote: Create a pull request for 'branch_1' on GitHub by visiting:
remote:   https://github.com/DenisRazvorotnikov/lab1_3/pull/new/branch_1
remote:
To github.com:DenisRazvorotnikov/lab1_3.git
 * [new branch]      branch_1 -> branch_1
denisrazvorotnikov@MacBook-Air-Denis lab1_3 %

```

Рисунок 11 – Отправка изменений на сервер

12. Создать средствами GitHub удаленную ветку branch_3.

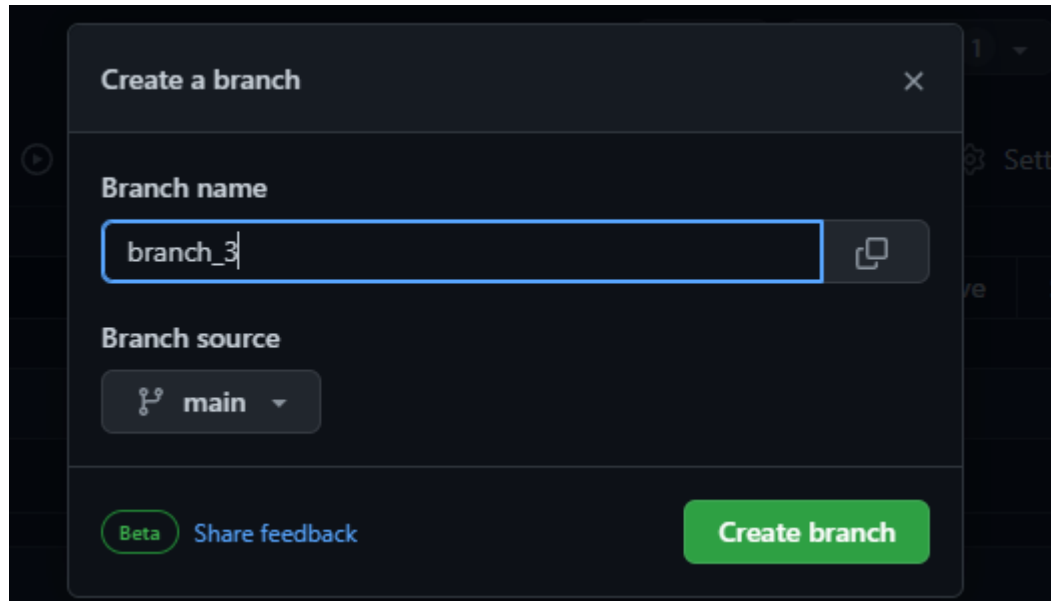


Рисунок 12 – Создание ветки на удаленном репозитории

13. Создать в локальном репозитории ветку отслеживания удаленной ветки branch_3.

```
denisrazvorotnikov@MacBook-Air-Denis lab1_3 % git checkout --track origin/branch_3
error: Your local changes to the following files would be overwritten by checkout:
  1.txt
  3.txt
Please commit your changes or stash them before you switch branches.
Aborting
denisrazvorotnikov@MacBook-Air-Denis lab1_3 %
```

Рисунок 13 – Создание ветки отслеживания branch_3

14. Перейти на ветку branch_3 и добавить файл 2.txt строку "the final fantasy in the 4.txt file".

```
C:\Users\ynakh\OneDrive\Рабочий стол\Git\Lab-3-OPJ>git commit
hint: Waiting for your editor to close the file... unix2dos: co
b-3-OPJ/.git/COMMIT_EDITMSG to DOS format...
dos2unix: converting file C:/Users/ynakh/OneDrive/Рабочий стол
[branch_3 1e8ec62] q
 1 file changed, 1 insertion(+)
 create mode 100644 2.txt
C:\Users\ynakh\OneDrive\Рабочий стол\Git\Lab-3-OPJ>
```

Рисунок 14 – Переход на ветку branch_3 и изменение файла

24. Отправить изменения веток master и branch_2 на GitHub.

```
C:\Users\ynakh\OneDrive\Рабочий стол\Git\Lab-3-ОПЭ>git push origin main
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Yana-Kh/Lab-3-ОПЭ.git
   7c00931..78f5da6  main -> main

C:\Users\ynakh\OneDrive\Рабочий стол\Git\Lab-3-ОПЭ>git push origin branch_2
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'branch_2' on GitHub by visiting:
remote:   https://github.com/Yana-Kh/Lab-3-ОПЭ/pull/new/branch_2
remote:
To https://github.com/Yana-Kh/Lab-3-ОПЭ.git
 * [new branch]      branch_2 -> branch_2
```

Рисунок 26 – Отправка изменений на GitHub.

Вопросы для защиты работы

1. Что такое ветка?

Ветка в Git — это просто легковесный подвижный указатель на один из КОММИТОВ.

2. Что такое HEAD?

HEAD – это указатель, задача которого ссылаться на определенный коммит в репозитории.

3. Способы создания веток.

С помощью команды `git branch` или `git checkout -b`

4. Как узнать текущую ветку?

С помощью команды `git branch`, напротив будет знак «*»

5. Как переключаться между ветками?

С помощью команды `git checkout <имя ветки>`

6. Что такое удаленная ветка?

Удалённые ветки — это ссылки на состояние веток в удаленных репозиториях.

7. Что такое ветка отслеживания?

Ветка отслеживания — это ссылка, расположенная локально, на определённое состояние удалённых веток.

8. Как создать ветку отслеживания?

Для синхронизации `git fetch origin`, а затем `git checkout --track origin/<название_ветки>`.

9. Как отправить изменения из локальной ветки в удаленную ветку?

С помощью команды `git push origin <branch >`

10. В чем отличие команд `git fetch` и `git pull`?

`Git pull` – это сочетание команд `git fetch` (получение изменений с удаленного репозитория) и `git merge` (объединение веток).

11. Как удалить локальную и удаленную ветки?

Удаление удаленной ветки: `git push origin --delete <branch >`

Удаление локальной: `git branch -d <branch >`

12. Изучить модель ветвления `git-flow` (использовать материалы статей <https://www.atlassian.com/ru/git/tutorials/comparing-workflows/gitflow-workflow>, <https://habr.com/ru/post/106912/>). Какие основные типы веток

присутствуют в модели git-flow? Как организована работа с ветками в модели git-flow? В чем недостатки git-flow?

Git-flow — альтернативная модель ветвления Git, в которой используются функциональные ветки и несколько основных веток.

Под каждую новую функцию нужно выделить собственную ветку, которую можно отправить в центральный репозиторий для создания резервной копии или совместной работы команды. Ветки feature создаются не на основе main, а на основе develop. Когда работа над функцией завершается, соответствующая ветка сливается с веткой develop. Функции не следует отправлять напрямую в ветку main.

Последовательность действий при работе по модели Gitflow:

1. Из ветки main создается ветка develop.
2. Из ветки develop создается ветка release.
3. Из ветки develop создаются ветки feature.
4. Когда работа над веткой feature завершается, она сливается в ветку develop.
5. Когда работа над веткой release завершается, она сливается с ветками develop и main.
6. Если в ветке main обнаруживается проблема, из main создается ветка hotfix.
7. Когда работа над веткой hotfix завершается, она сливается с ветками develop и main.

Первая проблема: авторам приходится использовать ветку develop вместо master, поскольку master зарезервирован для кода, который отправляется в продакшен. Существует сложившийся обычай называть рабочую ветвь по умолчанию master, и делать ответвления и слияния с ней. Большинство инструментов по умолчанию используют это название для основной ветки и по умолчанию выводят именно ее, и бывает неудобно постоянно переключаться вручную на другую ветку.

Вторая проблема процесса git flow – сложности, возникающие из-за веток для патчей и для релиза. Подобная структура может подойти некоторым организациям, но для абсолютного большинства она просто убийственно излишня. На сегодняшний день большинство компаний практикуют непрерывное развертывание (continuous delivery), что подразумевает, что основная ветвь по умолчанию может быть задеплоена (deploy). А значит, можно избежать использования веток для релиза и патчей, и всех связанных с ними хлопот, например, обратного слияния из веток релизов.