

Отчёт по лабораторной работе №4

Операционные системы

Лисенков Егор Романович

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Выводы	15
6	Список литературы	16

Список иллюстраций

Список таблиц

1 Цель работы

Научиться правильно работать с репозиториями git.

2 Задание

1. Выполнить работу для тестового репозитория.
2. Преобразовать рабочий репозиторий в репозиторий с git-flow и conventional commits.

3 Теоретическое введение

Gitflow Workflow опубликована и популяризована Винсентом Дриссеном. Gitflow Workflow предполагает выстраивание строгой модели ветвления с учётом выпуска проекта. Данная модель отлично подходит для организации рабочего процесса на основе релизов. Работа по модели Gitflow включает создание отдельной ветки для исправлений ошибок в рабочей среде. Последовательность действий при работе по модели Gitflow: Из ветки master создаётся ветка develop. Из ветки develop создаётся ветка release. Из ветки develop создаются ветки feature. Когда работа над веткой feature завершена, она сливается с веткой develop. Когда работа над веткой релиза release завершена, она сливается в ветки develop и master. Если в master обнаружена проблема, из master создаётся ветка hotfix. Когда работа над веткой исправления hotfix завершена, она сливается в ветки develop и master.

4 Выполнение лабораторной работы

Выполню загрузку из коллекции репозитория Corp. (рис. ??)

```
erlisenkov@dk3n65 # dnf copr enable elegos/gitflow
erlisenkov@dk3n65 # dnf install nodejs
Последняя проверка окончания срока действия метаданных: 0:01:20 назад, Чт 29 фев
2024 00:05:42.
Зависимости разрешены.
```

Выполню установку пакета Node.js, в которых находится программное обеспечение для выполнения 4 лабораторной работы. (рис. ??)

```
erlisenkov@dk3n65 :~# apt-get install pnpm
```

Далее выполню команду pnpm setup. (рис. ??)

```
erlisenkov@dk3n65 :~# pnpm setup
Appended new lines to /root/.bashrc

Next configuration changes were made:
export PNPM_HOME="/root/.local/share/pnpm"
case ":$PATH:" in
  *:$PNPM_HOME:* ) ;;
  *) export PATH="$PNPM_HOME:$PATH" ;;
esac

To start using pnpm, run:
source /root/.bashrc
```

Далее выполню команду, чтобы работать с окружением для git-flow. (рис. ??)

```
erlisenkov@dk3n65 :~$ source ~/.bashrc
```

Выполню команду чтобы помочь в форматировании коммитов. (рис. ??)


```

erlisenkov@dk3n65 :~$ pnpm add -g commitizen
Packages: +152
Progress: resolved 152, reused 0, downloaded 152, added 152, done
+ commitizen 4.3.0
Done in 40.8s

```

Далее выполняем команду, которая будет создавать логи. (рис. ??)

```

erlisenkov@dk3n65 :~$ pnpm add -g standard-changelog
Packages: +56
Progress: resolved 208, reused 152, downloaded 56, added 56, done
+ standard-changelog 5.0.0
Done in 43.4s

```

Далее мы создаём репозиторий git-etended на Git&hub. Тем самым следующей командой создаём соответствующий репозиторий на консоли и заходим туда. (рис. ??)

```

erlisenkov@dk3n65 :~/work$ mkdir git-extended
erlisenkov@dk3n65 :~/work$ cd git-extended/

```

Создадим первый файл README.md и выполним инициализацию репозитория. (рис. ??)

```

erlisenkov@dk3n65 :~/work/git-extended$ echo "# git-extended" >> README.md
erlisenkov@dk3n65 :~/work/git-extended$ ls
README.md
erlisenkov@dk3n65 :~/work/git-extended$ git init
подсказка: Using 'master' as the name for the initial branch. This default branch name
подсказка: is subject to change. To configure the initial branch name to use in all
подсказка: of your new repositories, which will suppress this warning, call:
подсказка:
подсказка:     git config --global init.defaultBranch <name>
подсказка:
подсказка: Names commonly chosen instead of 'master' are 'main', 'trunk' and
подсказка: 'development'. The just-created branch can be renamed via this command:
подсказка:
подсказка:     git branch -m <name>

```

Выполню первый коммит и подключу удалённый репозиторий с помощью консольной команды git remote. Следующим шагом отправим изменения в репозиторий на сайт. (рис. ??)

```
erlisenkov@dk3n65: ~/work/git-extended$ git commit -m "first commit"
[master (корневой коммит) ecd839f] first commit
1 file changed, 1 insertion(+)
create mode 100644 README.md
erlisenkov@dk3n65: ~/work/git-extended$ git branch -M main
erlisenkov@dk3n65: ~/work/git-extended$ git remote add origin https://github.com,
erlisenkov@dk3n65: ~/work/git-extended$ git push -u origin main
Перечисление объектов: 3, готово.
Подсчет объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 226 байтов | 113.00 КиБ/с, готово.
Всего 3 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано
```

Проверил что на сайте изменения были проведены. Далее создам файл package.json с помощью команды `pnpm init`. (рис. ??)

```
erlisenkov@dk3n65: ~/work/git-extended$ pnpm init
{
  "name": "git-extended",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}
erlisenkov@dk3n65: ~/work/git-extended$ ls
package.json  README.md
erlisenkov@dk3n65: ~/work/git-extended$
```

Выполню небольшие изменения в файле. (рис. ??)

```
"name": "git-extended",
"version": "1.0.0",
"description": "",
"main": "index.js",
```

Добавлю новые файлы (`git add`) и выполню коммит (`git cz`). (рис. ??)

```
erlisenkov@dk3n65: ~/work/git-extended$ git cz
cz-cli@4.3.0, cz-conventional-changelog@3.3.0

Select the type of change that you're committing: style:  Changes that do not affect the meaning of the code
white-space, formatting, missing semi-colons, etc)
What is the scope of this change (e.g. component or file name): (press enter to skip)
Write a short, imperative tense description of the change (max 93 chars):
(4) code
Provide a longer description of the change: (press enter to skip)

Are there any breaking changes? No
Does this change affect any open issues? No
[main fc95ef9] style: code
```

Отправлю изменения. (рис. ??)

```
erlisenkov@dk3n65: ~/work/git-extended$ git push
Перечисление объектов: 4, готово.
Подсчет объектов: 100% (4/4), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 492 байта | 492.00 КиБ/с, готово.
Всего 3 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
```

Для дальнейшей работы я инициализирую `git-flow` с `git flow init` (Префикс: `v`). (рис. ??)

```
erlisenkov@dk3n65 : ~/work/git-extended$ git flow init
Which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main] main
Branch name for "next release" development: [develop] develop

How to name your supporting branch prefixes?
Feature branches? [feature/] v
Bugfix branches? [bugfix/] c
Release branches? [release/] v
Hotfix branches? [hotfix/] v
Support branches? [support/] v
Version tag prefix? [] v
Hooks and filters directory? [/home/aifedorova/work/git-extended/.git/hooks] /home/aifedorova/work/git-extended/.git/hooks
```

Выполню проверку, что нахожусь в ветке develop. (рис. ??)

```
erlisenkov@dk3n65 : ~/work/git-extended$ git branch
* develop
main
```

Загружу весь репозиторий. (рис. ??)

```
erlisenkov@dk3n65 : ~/work/git-extended$ git push --all
Всего 0 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
```

Загрузим внешнюю ветку как вышестоящую. (рис. ??)

```
erlisenkov@dk3n65 : ~/work/git-extended$ git branch --set-upstream-to=origin/develop develop
branch 'develop' set up to track 'origin/develop'.
```

Создам релиз с версией 1.0.0. (рис. ??)

```
erlisenkov@dk3n65 : ~/work/git-extended$ git flow release start 1.0.0
Переключились на новую ветку «v1.0.0»

Summary of actions:
- A new branch 'v1.0.0' was created, based on 'develop'
- You are now on branch 'v1.0.0'

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'
```

Создадим журнал изменений. (рис. ??)

```
erlisenkov@dk3n65 : ~/work/git-extended$ standard-changelog --first-release
✓ created CHANGELOG.md
✓ output changes to CHANGELOG.md
```

Занесём в индекс журнал изменений. (рис. ??)

```
erlisenkov@dk3n65 : ~/work/git-extended$ git add CHANGELOG.md
erlisenkov@dk3n65 : ~/work/git-extended$ git commit -am 'chore(site): add changelog'
[v1.0.0 5064ca9] chore(site): add changelog
1 file changed, 4 insertions(+)
create mode 100644 CHANGELOG.md
```

Далее выполняем команды, которые позволят нам положить релизную ветку в основную. Введём сообщения, чтобы программа понимала цели изменения. (рис. ??)

```

Merge branch 'v1.0.0'
# Пожалуйста, введите сообщение коммита, для объяснения, зачем нужно
# это слияние, особенно, если это слияние обновленной вышестоящей
# ветки в тематическую ветку.
#
# Строки, начинающиеся с «#» будут проигнорированы, а пустое сообщение
# отменяет процесс коммита.

```

Успешное выполнение! (рис. ??)

```

erlisenkov@dk3n65 :~/work/git-extended$ git flow release finish 1.0.0
Переключились на ветку «main»
Эта ветка соответствует «origin/main».
Merge made by the 'ort' strategy.
  CHANGELOG.md | 4 +++++
  1 file changed, 4 insertions(+)
  create mode 100644 CHANGELOG.md
Уже на «main»
Ваша ветка опережает «origin/main» на 2 коммита.
(используйте «git push», чтобы опубликовать ваши локальные коммиты)
Переключились на ветку «develop»
Эта ветка соответствует «origin/develop».
warning: refname 'v1.0.0' is ambiguous.
warning: refname 'v1.0.0' is ambiguous.
Merge made by the 'ort' strategy.
  CHANGELOG.md | 4 +++++
  1 file changed, 4 insertions(+)
  create mode 100644 CHANGELOG.md
Ветка v1.0.0 удалена (была 5064ca9).

```

Синхронизирую с git. (рис. ??)

```

erlisenkov@dk3n65 :~/work/git-extended$ git push --all
Перечисление объектов: 6, готово.
Подсчет объектов: 100% (6/6), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (4/4), готово.
Запись объектов: 100% (5/5), 547 байтов | 547.00 КиБ/с, готово.
Всего 5 (изменений 2), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (2/2), done.

```

Синхронизируем все поставленные теги. (рис. ??)

```

erlisenkov@dk3n65 :~/work/git-extended$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 155 байтов | 155.00 КиБ/с, готово.
Всего 1 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0

```

А теперь создадим релиз на git&hub. (рис. ??)

```

erlisenkov@dk3n65 :~/work/git-extended$ gh release create v1.0.0 -F CHANGELOG.md

```

Дополнительно создам ветку для новой функциональности. (рис. ??)

```

erlisenkov@dk3n65 :~/work/git-extended$ git flow feature start feature_branch
Переключились на новую ветку «vfeature_branch»

Summary of actions:
- A new branch 'vfeature_branch' was created, based on 'develop'
- You are now on branch 'vfeature_branch'

Now, start committing on your feature. When done, use:

    git flow feature finish feature_branch

```

Выполню объединение веток. (рис. ??)

```
erlisenkov@dk3n65:~/work/git-extended$ git flow feature finish feature_branch
Переключились на ветку «develop»
Эта ветка соответствует «origin/develop».
Уже актуально.
Ветка vfeature_branch удалена (была 761002d).

Summary of actions:
- The feature branch 'vfeature_branch' was merged into 'develop'
- Feature branch 'vfeature_branch' has been locally deleted
- You are now on branch 'develop'
```

Создам релиз 1.2.3. (рис. ??)

```
erlisenkov@dk3n65:~/work/git-extended$ git flow release start 1.2.3
Переключились на новую ветку «v1.2.3»

Summary of actions:
- A new branch 'v1.2.3' was created, based on 'develop'
- You are now on branch 'v1.2.3'

Follow-up actions:
- Bump the version number now!
- Start committing last-minute fixes in preparing your release
- When done, run:

    git flow release finish '1.2.3'
```

Подредактирую файл package.json поменяя версию на 1.2.3. (рис. ??)

```
package.json  [-M--] 19 L: [ 1+ 7  8/ 15] *(231 / 342b) 0034 0x022
{
  "name": "git-extended",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
```

Выполню создание журнала изменений. (рис. ??)

```
erlisenkov@dk3n65:~/work/git-extended$ standard-changelog
✓ output changes to CHANGELOG.md
erlisenkov@dk3n65:~/work/git-extended$
```

Добавим журнал изменений в индекс. (рис. ??)

```
erlisenkov@dk3n65:~/work/git-extended$ git add CHANGELOG.md
erlisenkov@dk3n65:~/work/git-extended$ git commit -am 'chore(site): update changelog'
[v1.2.3 42a0719] chore(site): update changelog
1 file changed, 1 insertion(+), 1 deletion(-)
```

Выполню перенаправление релизной ветки в основную ветку и отвечу на важные вопросы. (рис. ??)

```
Merge tag 'v1.2.3' into develop

v
# Пожалуйста, введите сообщение коммита, для объяснения, зачем нужно
# это слияние, особенно, если это слияние обновленной вышестоящей
# ветки в тематическую ветку.
#
# Строки, начинающиеся с «#» будут проигнорированы, а пустое сообщение
# отменяет процесс коммита.
```

Всё прошло успешно! (рис. ??)

```
erlisenkov@dk3n65 :~/work/git-extended$ git flow release finish 1.2.3
Переключились на ветку «main»
Эта ветка соответствует «origin/main».
Merge made by the 'ort' strategy.
 package.json | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
Уже на «main»
Ваша ветка опережает «origin/main» на 3 коммита.
(используйте «git push», чтобы опубликовать ваши локальные коммиты)
Переключились на ветку «develop»
Эта ветка соответствует «origin/develop».
warning: refname 'v1.2.3' is ambiguous.
warning: refname 'v1.2.3' is ambiguous.
Merge made by the 'ort' strategy.
 package.json | 2 +-

```

Синхронизация с git. (рис. ??)

```
erlisenkov@dk3n65 :~/work/git-extended$ git push --all
Перечисление объектов: 7, готово.
Подсчет объектов: 100% (7/7), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (5/5), готово.
Запись объектов: 100% (5/5), 581 байт | 581.00 КиБ/с, готово.
Всего 5 (изменений 3), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (3/3), completed with 2 local objects.

```

Синхронизация тегов на git&hub. (рис. ??)

```
erlisenkov@dk3n65 :~/work/git-extended$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 155 байтов | 155.00 КиБ/с, готово.
Всего 1 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
* [new tag]          v1.2.3 -> v1.2.3

```

Создание релиза с комментарием из журнала изменений (на git.hub) . (рис. ??)

```
erlisenkov@dk3n65 :~/work/git-extended$ gh release create v1.2.3 -F CHANGELOG.md

```

Все изменения были приняты и всё выполнилось!!!! (рис. ??)

CHANGELOG.md	chore(site): add changelog
README.md	first commit
package.json	chore(site): update changelog

5 Выводы

Я получил самые важные знания для работы с git и более запомнил материал!

6 Список литературы

Лабораторная работа №4