

Отчёт по лабораторной работе 4

Продвинутое использование git

Скрипник Виктория Дмитриевна

Содержание

1	Цель работы	5
2	Теоретические сведения	6
3	Выполнение лабораторной работы	7
3.1	Работа с тестовым репозиторием	7
3.2	Подготовка рабочего репозитория	18
4	Вывод	20

Список иллюстраций

3.1	Node.js	8
3.2	установка commitizen	9
3.3	установка standard-changelog	10
3.4	package.json	11
3.5	Отправка	12
3.6	Инициализация и загрузка изменений	13
3.7	Завершение релиза	14
3.8	Отправка	15
3.9	Объединение веток	16
3.10	Завершение релиза	17
3.11	package.json и коммит	18
3.12	Завершение релиза	19

Список таблиц

1 Цель работы

Получение навыков правильной работы с репозиториями git.

2 Теоретические сведения

- Gitflow Workflow опубликована и популяризована Винсентом Дриссенем.
- 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.

3 Выполнение лабораторной работы

3.1 Работа с тестовым репозиторием

Для работы с Node.js добавим каталог с исполняемыми файлами, устанавливаемыми yarn, в переменную PATH.

```
vdskripnik@vdskripnik:~$  
vdskripnik@vdskripnik:~$ pnpm setup  
Appended new lines to /home/vdskripnik/.bashrc  
  
Next configuration changes were made:  
export PNPM_HOME="/home/vdskripnik/.local/share/pnpm"  
case "$PATH:" in  
  *:$PNPM_HOME:*) ;;  
  *) export PATH="$PNPM_HOME:$PATH" ;;  
esac  
  
To start using pnpm, run:  
source /home/vdskripnik/.bashrc  
vdskripnik@vdskripnik:~$ source ~/.bashrc  
vdskripnik@vdskripnik:~$
```

Рис. 3.1: Node.js

Программа `commitizen` используется для помощи в форматировании коммитов. При этом устанавливается скрипт `git-cz`, который мы и будем использовать для коммитов.


```
vdscriprik@vdscriprik:~$  
vdscriprik@vdscriprik:~$ pnpm add -g commitizen  
  
Update available! 9.13.0 → 10.12.1.  
Changelog: https://github.com/pnpm/pnpm/releases/tag/v10.12.1  
Run "pnpm self-update" to update.  
  
Follow @pnpmjs for updates: https://x.com/pnpmjs  
  
WARN 2 deprecated subdependencies found: glob@7.2.3, inflight@1.0.6  
Packages: +151  
+++++  
Progress: resolved 151, reused 0, downloaded 151, added 151, done  
  
/home/vdscriprik/.local/share/pnpm/global/5:  
+ commitizen 4.3.1  
  
Done in 3.2s  
vdscriprik@vdscriprik:~$
```

Рис. 3.2: установка commitizen

Программа standard-changelog используется для помощи в создании логов.

```
vdskripnik@vdskripnik:~$  
vdskripnik@vdskripnik:~$ pnpm add -g standard-changelog  
WARN 2 deprecated subdependencies found: glob@7.2.3, inflight@1.0.6  
Packages: +34  
++++  
Progress: resolved 185, reused 151, downloaded 34, added 34, done  
  
/home/vdskripnik/.local/share/pnpm/global/5:  
+ standard-changelog 7.0.1  
  
Done in 3.7s  
vdskripnik@vdskripnik:~$
```

Рис. 3.3: установка standard-changelog

Делаем первый коммит и выкладываем на github.

Необходимо заполнить несколько параметров пакета.

Таким образом, файл package.json приобретает вид:



Рис. 3.4: package.json

Добавим новые файлы.

Выполним коммит.

Отправим на github.

```

vdskrprik@vdskrprik: ~/git-extended$ git add .
vdskrprik@vdskrprik: ~/git-extended$ git cz
cz-cli@4.3.1, cz-conventional-changelog@3.3.0

? Select the type of change that you're committing: chore:  Other changes that don't modify src or test files
? 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) test
? 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
[master a5d0d6f] chore: test
1 file changed, 14 insertions(+)
create mode 100644 package.json
vdskrprik@vdskrprik: ~/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), 1.11 КиБ | 1.11 МБ/с, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:vdskrprik/git-extended.git
  52e1be9..a5d0d6f  master -> master
vdskrprik@vdskrprik: ~/git-extended$

```

Рис. 3.5: Отправка

Инициализируем git-flow

Проверьте, что Вы на ветке develop

Загрузите весь репозиторий в хранилище

```

vdscripnik@vdscripnik:~/git-extended$
vdscripnik@vdscripnik:~/git-extended$ 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? []
Hooks and filters directory? [/home/vdscripnik/git-extended/.git/hooks]
vdscripnik@vdscripnik:~/git-extended$ git branch
* develop
  master

vdscripnik@vdscripnik:~/git-extended$ git push --all
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:   https://github.com/vdscripnik/git-extended/pull/new/develop
remote:
To github.com:vdscripnik/git-extended.git
 * [new branch]      develop -> develop
vdscripnik@vdscripnik:~/git-extended$

```

Рис. 3.6: Инициализация и загрузка изменений

Установите внешнюю ветку как вышестоящую для этой ветки

Создадим релиз с версией 1.0.0

Создадим журнал изменений

Добавим журнал изменений в индекс

Зальём релизную ветку в основную ветку

```

vdscripnik@vdscripnik:~/git-extended$ git branch --set-upstream-to=origin/develop develop
branch 'develop' set up to track 'origin/develop'.
vdscripnik@vdscripnik:~/git-extended$ git flow release start 1.0.0
Переключились на новую ветку «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'

vdscripnik@vdscripnik:~/git-extended$ standard-changelog --first-release
vdscripnik@vdscripnik:~/git-extended$ git add CHANGELOG.md
vdscripnik@vdscripnik:~/git-extended$ git commit -am 'chore(site): add changelog'
[release/1.0.0 ec5feb2] chore(site): add changelog
1 file changed, 4 insertions(+)
create mode 100644 CHANGELOG.md
vdscripnik@vdscripnik:~/git-extended$ git flow release finish 1.0.0
Переключились на ветку «master»
Эта ветка соответствует «origin/master».
Merge made by the 'ort' strategy.
CHANGELOG.md | 4 +++
1 file changed, 4 insertions(+)
create mode 100644 CHANGELOG.md
Уже на «master»
Ваша ветка опережает «origin/master» на 2 коммита.
(используйте «git push», чтобы опубликовать ваши локальные коммиты)
Переключились на ветку «develop»
Эта ветка соответствует «origin/develop».
Merge made by the 'ort' strategy.

```

Рис. 3.7: Завершение релиза

Отправим данные на github

```
vdscripnik@vdscripnik:~/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), 2.66 КиБ | 2.67 МиБ/с, готово.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:vdscripnik/git-extended.git
   a5d0d6f..01a47fb develop -> develop
   a5d0d6f..91caf2d master -> master
vdscripnik@vdscripnik:~/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.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:vdscripnik/git-extended.git
   * [new tag]         1.0.0 -> 1.0.0
vdscripnik@vdscripnik:~/git-extended$ gh release create v1.0.0 -F CHANGELOG.md
https://github.com/vdscripnik/git-extended/releases/tag/v1.0.0
vdscripnik@vdscripnik:~/git-extended$
```

Рис. 3.8: Отправка

Создадим ветку для новой функциональности По окончании разработки новой функциональности следующим шагом следует объединить ветку `feature_branch` с `develop`:

```

vdscripnik@vdscripnik:~/git-extended$ git flow feature start feature_branch
Переключились на новую ветку «feature/feature_branch»

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

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

    git flow feature finish feature_branch

vdscripnik@vdscripnik:~/git-extended$ touch test2
vdscripnik@vdscripnik:~/git-extended$ git flow feature finish feature_branch
Переключились на ветку «develop»
Эта ветка соответствует «origin/develop».
Уже актуально.
Ветка feature/feature_branch удалена (была 01a47fb).

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

vdscripnik@vdscripnik:~/git-extended$

```

Рис. 3.9: Объединение веток

Создадим релиз с версией 1.2.3

Обновите номер версии в файле package.json. Установите её в 1.2.3

Создадим журнал изменений

Добавим журнал изменений в индекс

Зальём релизную ветку в основную ветку


```
vdscripnik@vdscripnik:~/git-extended$ git push --all
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 2.77 КиБ | 2.77 МБ/с, готово.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:vdscripnik/git-extended.git
   01a47fb..60ee323  develop -> develop
   91caf2d..8e203f0  master -> master
vdscripnik@vdscripnik:~/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.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:vdscripnik/git-extended.git
 * [new tag]         1.2.3 -> 1.2.3
vdscripnik@vdscripnik:~/git-extended$ gh release create v1.2.3 -F CHANGELOG.md
https://github.com/vdscripnik/git-extended/releases/tag/v1.2.3
vdscripnik@vdscripnik:~/git-extended$
```

Рис. 3.10: Завершение релиза

3.2 Подготовка рабочего репозитория

```
vdskripnik@vdskripnik:~/work/study/2024-2025/Операционные системы/os-intro$ git add .
vdskripnik@vdskripnik:~/work/study/2024-2025/Операционные системы/os-intro$ git cz
cz-cli@4.3.1, cz-conventional-changelog@3.3.0

? Select the type of change that you're committing: chore:  Other changes that don't modify src or test files
? 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) test
? 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
[master 8aa4482] chore: test
1 file changed, 15 insertions(+)
create mode 100644 package.json
vdskripnik@vdskripnik:~/work/study/2024-2025/Операционные системы/os-intro$ 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), 1.11 КиБ | 1.11 МБ/с, готово.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:vdskripnik/os-intro.git
   daad5a1..8aa4482  master -> master
vdskripnik@vdskripnik:~/work/study/2024-2025/Операционные системы/os-intro$
```

Рис. 3.11: package.json и коммит

```
- Release branch 'release/1.0.0' has been locally deleted
- You are now on branch 'develop'

vdscripnik@vdscripnik:~/work/study/2024-2025/Операционные системы/os-intro$ 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), 2.72 КиБ | 2.72 МБ/с, готово.
Total 5 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:vdscripnik/os-intro.git
   8aa4482..179d702  develop -> develop
   8aa4482..a4c3eab  master -> master

vdscripnik@vdscripnik:~/work/study/2024-2025/Операционные системы/os-intro$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 156 байтов | 156.00 КиБ/с, готово.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:vdscripnik/os-intro.git
 * [new tag]         1.0.0 -> 1.0.0
vdscripnik@vdscripnik:~/work/study/2024-2025/Операционные системы/os-intro$
```

Рис. 3.12: Завершение релиза

4 Вывод

Мы приобрели практические навыки взаимодействия с дополнительными функциями гитхаб.