

Отчёт по лабораторной работе 4

Продвинутое использование git

Шамес Эддин Хамза

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.

```
hamzaru@hamzaru:~$ pnpm setup
Appended new lines to /home/hamzaru/.bashrc

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

To start using pnpm, run:
source /home/hamzaru/.bashrc
hamzaru@hamzaru:~$
```

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.


```
hamzaru@hamzaru:~$ 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/hamzaru/.local/share/pnpm/global/5:
+ commitizen 4.3.1

Done in 3.9s
hamzaru@hamzaru:~$
```

Рис. 3.2: установка commitizen

Программа standard-changelog используется для помощи в создании логов.

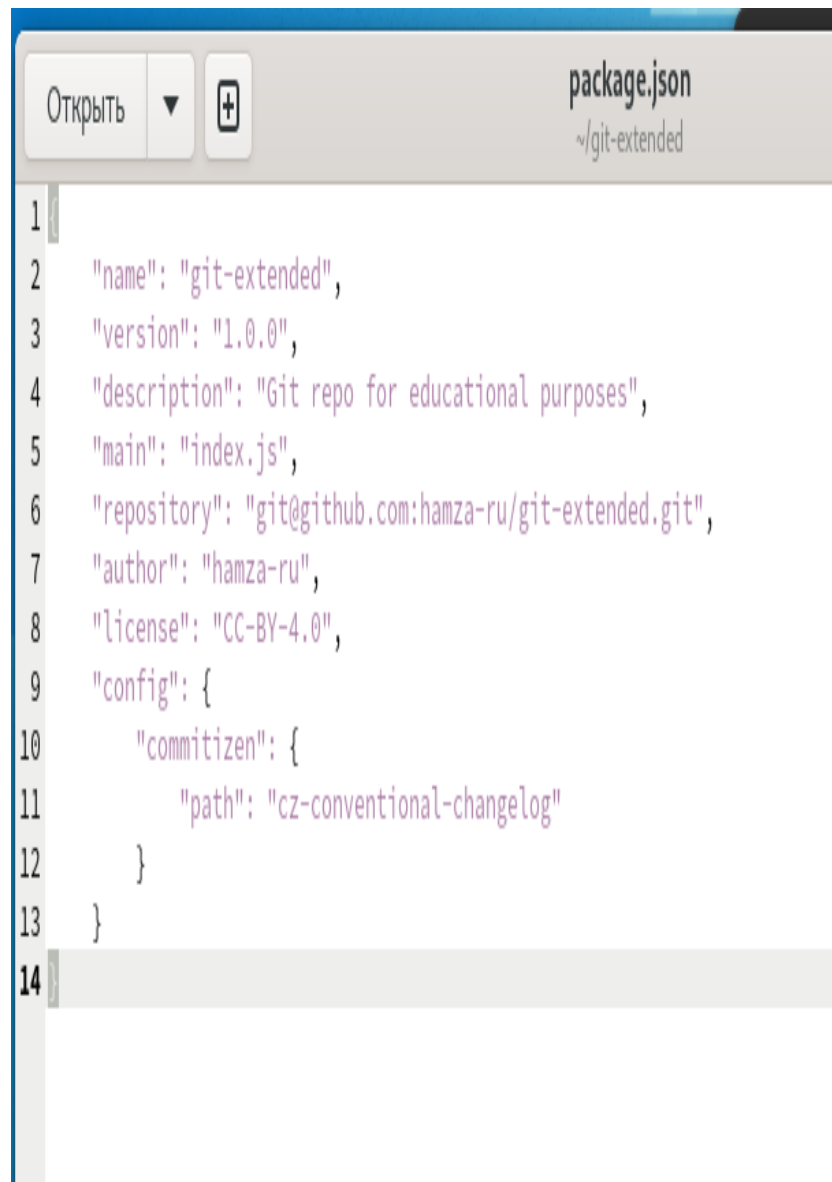
```
hamzaru@hamzaru:~$  
hamzaru@hamzaru:~$ pnpm add -g standard-changelog  
WARN 2 deprecated subdependencies found: glob@7.2.3, inflight@1.0.6  
Packages: +34  
++++  
Progress: resolved 185, reused 151, downloaded 34, added 34, done  
  
/home/hamzaru/.local/share/pnpm/global/5:  
+ standard-changelog 7.0.1  
  
Done in 4.1s  
hamzaru@hamzaru:~$
```

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:



The image shows a code editor window with a title bar that includes a button labeled 'Открыть' (Open), a dropdown arrow, and a file icon. The file name 'package.json' and its path '~/.git-extended' are displayed on the right. The editor contains 14 lines of JSON code, which is a package.json file. The code is as follows:

```
1  
2   "name": "git-extended",  
3   "version": "1.0.0",  
4   "description": "Git repo for educational purposes",  
5   "main": "index.js",  
6   "repository": "git@github.com:hamza-ru/git-extended.git",  
7   "author": "hamza-ru",  
8   "license": "CC-BY-4.0",  
9   "config": {  
10     "commitizen": {  
11       "path": "cz-conventional-changelog"  
12     }  
13   }  
14 }
```

Рис. 3.4: package.json

Добавим новые файлы.

Выполним коммит.

Отправим на github.

```

hamzaru@hamzaru:~/git-extended$ git add .
hamzaru@hamzaru:~/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 ab89ee7] chore: test
1 file changed, 14 insertions(+)
create mode 100644 package.json
hamzaru@hamzaru:~/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:hamza-ru/git-extended.git
107391e..ab89ee7 master -> master
hamzaru@hamzaru:~/git-extended$

```

Рис. 3.5: Отправка

Инициализируем git-flow

Проверьте, что Вы на ветке develop

Загрузите весь репозиторий в хранилище

```

hamzaru@hamzaru:~/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/hamzaru/git-extended/.git/hooks]
hamzaru@hamzaru:~/git-extended$ git branch
* develop
  master
hamzaru@hamzaru:~/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/hamza-ru/git-extended/pull/new/develop
remote:
To github.com:hamza-ru/git-extended.git
* [new branch]      develop -> develop
hamzaru@hamzaru:~/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

```

hamzaru@hamzaru:~/git-extended$ standard-changelog --first-release
hamzaru@hamzaru:~/git-extended$ git add CHANGELOG.md
hamzaru@hamzaru:~/git-extended$ git commit -am 'chore(site): add changelo
g'
[release/1.0.0 99f2c78] chore(site): add changelog
1 file changed, 4 insertions(+)
create mode 100644 CHANGELOG.md
hamzaru@hamzaru:~/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.
CHANGELOG.md | 4 +++
1 file changed, 4 insertions(+)
create mode 100644 CHANGELOG.md
Ветка release/1.0.0 удалена (была 99f2c78).

```

Рис. 3.7: Завершение релиза

Отправим данные на github

```

hamzaru@hamzaru:~/git-extended$
hamzaru@hamzaru:~/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 | 2.66 МиБ/с, готово.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:hamza-ru/git-extended.git
   ab89ee7..5a5e32b  develop -> develop
   ab89ee7..2ceaf69  master -> master
hamzaru@hamzaru:~/git-extended$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 153 байта | 153.00 КиБ/с, готово.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:hamza-ru/git-extended.git
   * [new tag]         1.0.0 -> 1.0.0
hamzaru@hamzaru:~/git-extended$ gh release create v1.0.0 -F CHANGELOG.md
https://github.com/hamza-ru/git-extended/releases/tag/v1.0.0
hamzaru@hamzaru:~/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`:

```

hamzaru@hamzaru:~/git-extended$
hamzaru@hamzaru:~/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

hamzaru@hamzaru:~/git-extended$ touch test2
hamzaru@hamzaru:~/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 5a5e32b).

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'

hamzaru@hamzaru:~/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


```

Summary of actions:
- Release branch 'release/1.2.3' has been merged into 'master'
- The release was tagged '1.2.3'
- Release tag '1.2.3' has been back-merged into 'develop'
- Release branch 'release/1.2.3' has been locally deleted
- You are now on branch 'develop'

hamzaru@hamzaru:~/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:hamza-ru/git-extended.git
   5a5e32b..a37d824  develop -> develop
   2ceaf69..377a0b2  master -> master
hamzaru@hamzaru:~/git-extended$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 152 байта | 152.00 КиБ/с, готово.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:hamza-ru/git-extended.git
   * [new tag]         1.2.3 -> 1.2.3
hamzaru@hamzaru:~/git-extended$ gh release create v1.2.3 -F CHANGELOG.md
https://github.com/hamza-ru/git-extended/releases/tag/v1.2.3
hamzaru@hamzaru:~/git-extended$

```

Рис. 3.10: Завершение релиза

3.2 Подготовка рабочего репозитория

```
? 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 57547dd] chore: test
1 file changed, 14 insertions(+)
create mode 100644 package.json
hamzaru@hamzaru:~/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), 1.10 КиБ | 1.10 МиБ/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:hamza-ru/os-intro.git
ca6ee29..57547dd master -> master
hamzaru@hamzaru:~/work/study/2024-2025/Операционные системы/os-intro$
```

Рис. 3.11: package.json и коммит

```
Summary of actions:
- Release branch 'release/1.0.0' has been merged into 'master'
- The release was tagged '1.0.0'
- Release tag '1.0.0' has been back-merged into 'develop'
- Release branch 'release/1.0.0' has been locally deleted
- You are now on branch 'develop'

hamzaru@hamzaru:~/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), 2.77 КиБ | 2.77 МиБ/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:hamza-ru/os-intro.git
   57547dd..d15856d  develop -> develop
   57547dd..b5a1762  master -> master
hamzaru@hamzaru:~/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), 153 байта | 153.00 КиБ/с, готово.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:hamza-ru/os-intro.git
 * [new tag]         1.0.0 -> 1.0.0
hamzaru@hamzaru:~/work/study/2024-2025/Операционные системы/os-intro$
```

Рис. 3.12: Завершение релиза

4 Вывод

Мы приобрели практические навыки взаимодействия с дополнительными функциями гитхаб.