

Отчёт по лабораторной работе №4

Симонова Виктория Игоревна

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
4.1	Установка git-flow	8
4.2	Установка Node.js	8
4.3	Настройка Node.js	9
4.4	Общепринятые коммиты	9
5	Выводы	19
	Список литературы	20

Список иллюстраций

4.1	Установка git-flow	8
4.2	Установка Node.js	8
4.3	Установка Node.js	9
4.4	Настройка Node.js	9
4.5	Настройка Node.js	9
4.6	Установка программы	9
4.7	Установка программы	10
4.8	Создание репозитория	10
4.9	Создание репозитория	11
4.10	Конфигурация общепринятых коммитов	11
4.11	Изменяю файл	11
4.12	Отправка на github	12
4.13	Инициализация	12
4.14	Загрузка репозитория	12
4.15	Установка внешней ветки	13
4.16	Сливаю ветки	13
4.17	Комментарии	13
4.18	Завершение	14
4.19	Отправка файлов	14
4.20	Готовый релиз	15
4.21	Новая ветка	15
4.22	Новый релиз	16
4.23	Новый номер релиза	16
4.24	Новый номер релиза	17
4.25	Новый номер релиза	17
4.26	Готовый релиз	18

Список таблиц

1 Цель работы

Получение навыков правильной работы с репозиториями git. Конкретно в данной лабораторной работе мы работаем с gitflow это определённая надстройка над моделью git.

2 Задание

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

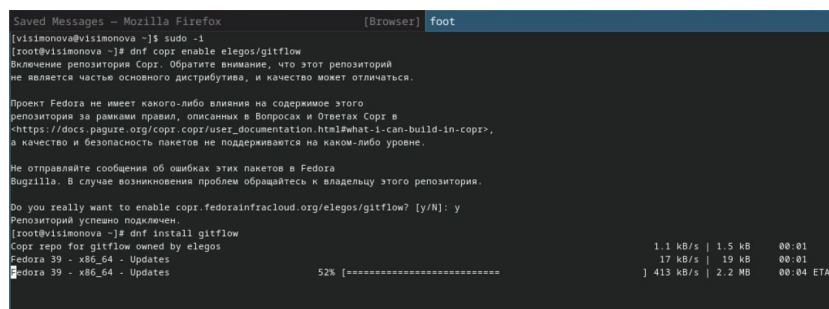
3 Теоретическое введение

- 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

4.1 Установка git-flow

Установка из коллекции репозитория Copr (рис. 4.1).



```

Saved Messages - Mozilla Firefox [Browser] foot
[visimonova@visimonova ~]$ sudo -i
[root@visimonova ~]# dnf copr enable elegos/gitflow
Включение репозитория Copr. Обратите внимание, что этот репозиторий
не является частью основного дистрибутива, и качество может отличаться.

Проект Fedora не имеет какого-либо влияния на содержимое этого
репозитория за рамками правил, описанных в Вопросах и Ответах Copr в
<https://docs.pagure.org/copr.copr/user_documentation.html#what-i-can-build-in-copr>,
а качество и безопасность пакетов не поддерживается на каком-либо уровне.

Не отправляйте сообщения об ошибках этих пакетов в Fedora
Bugzilla. В случае возникновения проблем обращайтесь к владельцу этого репозитория.

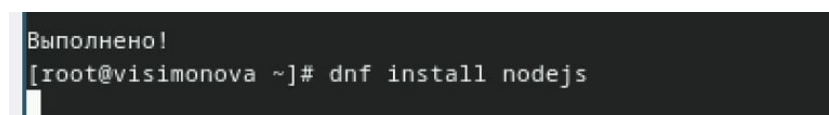
Do you really want to enable copr.fedorainfracloud.org/elegos/gitflow? [y/N]: y
[root@visimonova ~]# dnf install gitflow
Copr repo for gitflow owned by elegos
Fedora 39 - x86_64 - Updates 1.1 kB/s | 1.5 kB 00:01
Fedora 39 - x86_64 - Updates 17 kB/s | 19 kB 00:01
Fedora 39 - x86_64 - Updates 52% [=====] 413 kB/s | 2.2 MB 00:04 ETA

```

Рис. 4.1: Установка git-flow

4.2 Установка Node.js

На Node.js базируется программное обеспечение для семантического версионирования и общепринятых коммитов.(рис. 4.2).



```

Выполнено!
[root@visimonova ~]# dnf install nodejs

```

Рис. 4.2: Установка Node.js

Выполняю команды (рис. 4.3).


```
[root@visimonova ~]# apt-get install pnpm
```

Рис. 4.3: Установка Node.js

4.3 Настройка Node.js

Для работы с Node.js добавляю каталог с исполняемыми файлами, устанавливаемыми yarn, в переменную PATH. Запускаю установку (рис. 4.4).

```
[root@visimonova ~]# 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
[root@visimonova ~]#
```

Рис. 4.4: Настройка Node.js

Выполняю команду `source ~/.bashrc` (рис. 4.5).

```
[root@visimonova ~]# source ~/.bashrc
```

Рис. 4.5: Настройка Node.js

4.4 Общепринятые коммиты

Устанавливаю commitizen. Данная программа используется для помощи в форматировании коммитов. (скрипт `git cz`) (рис. 4.6).

```
root@visimonova ~]# pnpm add -g commitizen
Packages: +152
Progress: resolved 152, reused 0, downloaded 152, added 152, done
/root/.local/share/pnpm/global/5:
+ commitizen 4.3.0
Done in 13.7s
```

Рис. 4.6: Установка программы

Устанавливаю standard-changelog. Данная программа используется для помощи в создании логов.(рис. 4.7).

```
[root@visimonova ~]# pnpm add -g standard-changelog
Packages: +56
Progress: resolved 208, reused 152, downloaded 56, added 56, done

/root/.local/share/pnpm/global/5:
+ standard-changelog 5.0.0

Done in 9.4s
```

Рис. 4.7: Установка программы

Создаю новый git репозиторий (рис. 4.8).

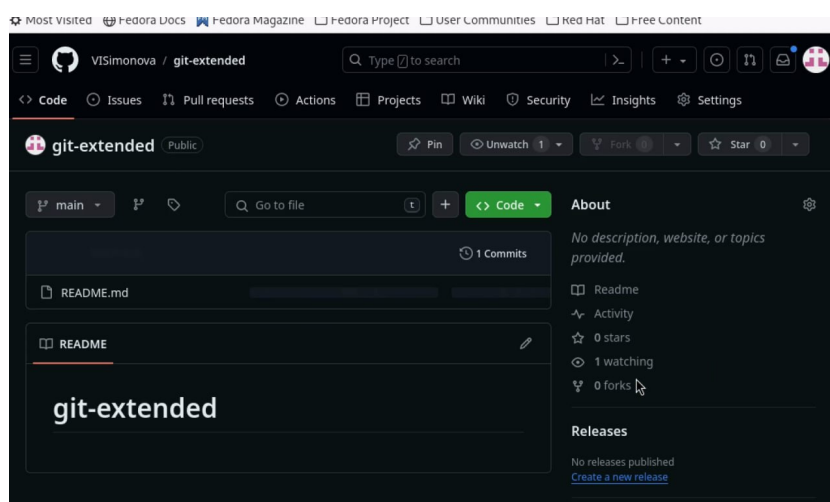


Рис. 4.8: Создание репозитория

Клонирую репозиторий к себе на компьютер делаю первый коммит и выкладываю на github (рис. 4.9).

```
[visimonova@visimonova ~]$ git clone --recursive git@github.com:VISimonova/git-extended.git
Клонирование в «git-extended»...
Enter passphrase for key '/home/visimonova/.ssh/id_rsa':
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Получение объектов: 100% (3/3), готово.
[visimonova@visimonova ~]$ mc

[visimonova@visimonova git-extended]$ git >> README.md
[visimonova@visimonova git-extended]$ git add README.md
[visimonova@visimonova git-extended]$ git commit -m "first commit"
[main c914c07] first commit
1 file changed, 45 insertions(+)
create mode 100644 README.md
[visimonova@visimonova git-extended]$ git remote add origin git@github.com:VISimonova/git-extended.git
error: внешний репозиторий origin уже существует
[visimonova@visimonova git-extended]$ git push -u origin master
error: src refspec master ничему не соответствует
error: не удалось отправить некоторые ссылки в «github.com:VISimonova/git-extended.git»
[visimonova@visimonova git-extended]$ git push -u origin main
Enter passphrase for key '/home/visimonova/.ssh/id_rsa':
Перечисление объектов: 4, готово.
Подсчет объектов: 100% (4/4), готово.
При скатии изменения используется до 4 потоков
Сжатие объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 2.30 KiB | 2.30 MiB/c, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:VISimonova/git-extended.git
 a1df100..c914c07  main -> main
branch 'main' set up to track 'origin/main'
```

Рис. 4.9: Создание репозитория

Конфигурация для пакетов Node.js (рис. 4.10).

```
[visimonova@visimonova git-extended]$ npm init
Wrote to /home/visimonova/git-extended/package.json

{
  "name": "git-extended",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}
```

Рис. 4.10: Конфигурация общепринятых коммитов

Изменяю файл package.json (рис. 4.11).

```
package.json  [-M--] 34 L: [ 1+ 6 7/ 14] *(230 / 370b) 0034 0x022

{
  "name": "git-extended",
  "version": "1.0.0",
  "description": "git-extended for educational purposes",
  "main": "index.js",
  "repository": "git@github.com:visimonova/git-extended.git",
  "author": "visimonova@visimonova",
  "license": "CC-BY-4.0",
  "scripts": {
    "commitizen": {
      "path": "c:/conventional-changelog"
    }
  }
}
```

Рис. 4.11: Изменяю файл

Добавляю новые файлы выполняю коммит и отправляю на github (рис. 4.12).

```

[visimonova@visimonova git-extended]$ git add .
[visimonova@visimonova git-extended]$ git cz
cz-cli@4.3.0, cz-conventional-changelog@3.3.0

? Select the type of change that you're committing: feat:      A new feature
? 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 94 chars):
(7) changel
? Provide a longer description of the change: (press enter to skip)

? Are there any breaking changes? Yes
? A BREAKING CHANGE commit requires a body. Please enter a longer description of the commit itself:
changel
? Describe the breaking changes:
changel
? Does this change affect any open issues? Yes
? Add issue references (e.g. "fix #123", "re #123").:

[main 0a3eac6] feat: changel
1 file changed, 14 insertions(+)
create mode 100644 package.json
[visimonova@visimonova git-extended]$ git push
Enter passphrase for key '/home/visimonova/.ssh/id_rsa':
Перечисление объектов: 4, готово.
Подсчет объектов: 100% (4/4), готово.
При скачивании изменений используется до 4 потоков
Сканирование объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 1.19 KiB | 1.19 МБ/с, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:VISimonova/git-extended.git
 c91dc07..0a3eac6  main -> main

```

Рис. 4.12: Отправка на github

Инициализируем git-flow (рис. 4.13).

```

[visimonova@visimonova git-extended]$ git flow init

Which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main]
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? [] v
Hooks and filters directory? [/home/visimonova/git-extended/.git/hooks]

```

Рис. 4.13: Инициализация

Проверяю ветку и загружаю репозиторий в хранилище (рис. 4.14).

```

[visimonova@visimonova git-extended]$ git branch
* develop
main
[visimonova@visimonova git-extended]$ git push --all
Enter passphrase for key '/home/visimonova/.ssh/id_rsa':
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/VISimonova/git-extended/pull/new/develop
remote:
To github.com:VISimonova/git-extended.git
 * [new branch]      develop -> develop

```

Рис. 4.14: Загрузка репозитория

Устанавливаю внешнюю ветку как вышестоящую для этой ветки. Создаю релиз с версией 1.0.0. Создаю журнал изменений. Добавляю журнал изменений

в индекс (рис. 4.15).

```
[visimonova@visimonova git-extended]$ git branch --set-upstream-to=origin/develop develop
branch 'develop' set up to track 'origin/develop'.
[visimonova@visimonova 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'

[visimonova@visimonova git-extended]$ standard-changelog --first-release
✓ created CHANGELOG.md
✓ output changes to CHANGELOG.md
[visimonova@visimonova git-extended]$ git add CHANGELOG.md
[visimonova@visimonova git-extended]$ git commit -am 'chore(site): add changelog'
[release/1.0.0 ea542aa] chore(site): add changelog
1 file changed, 14 insertions(+)
create mode 100644 CHANGELOG.md
```

Рис. 4.15: Установка внешней ветки

Заливаю релизную ветку в основную (рис. 4.16).

```
[visimonova@visimonova git-extended]$ git flow release finish 1.0.0
```

Рис. 4.16: Сливаю ветки

Обозначаю цель изменений (рис. 4.17).

```
Merge tag 'v1.0.0' into develop

releasel
# 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.
```

Рис. 4.17: Комментарии

Вижу сообщение об успешном завершении процесса (рис. 4.18).

```
[visimonova@visimonova 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 | 14 ++++++++
  1 file changed, 14 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».
Merge made by the 'ort' strategy.
  CHANGELOG.md | 14 ++++++++
  1 file changed, 14 insertions(+)
  create mode 100644 CHANGELOG.md
Ветка release/1.0.0 удалена (была ea542aa).

Summary of actions:
- Release branch 'release/1.0.0' has been merged into 'main'
- The release was tagged 'v1.0.0'
- Release tag 'v1.0.0' has been back-merged into 'develop'
- Release branch 'release/1.0.0' has been locally deleted
- You are now on branch 'develop'

[visimonova@visimonova git-extended]$
```

Рис. 4.18: Завершение

Отправляю данные на github и создаю релиз (рис. 4.19).

```
[visimonova@visimonova git-extended]$ git push --all
Enter passphrase for key '/home/visimonova/.ssh/id_rsa':
Перечисление объектов: 6, готово.
Подсчет объектов: 100% (6/6), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (5/5), готово.
Запись объектов: 100% (5/5), 2.87 КиБ | 2.87 МБ/с, готово.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:VISimonova/git-extended.git
   0a3eac6..e18f26a  develop -> develop
   0a3eac6..6ab509a  main -> main
[visimonova@visimonova git-extended]$ git push --tags
Enter passphrase for key '/home/visimonova/.ssh/id_rsa':
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 169 байтов | 169.00 КиБ/с, готово.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:VISimonova/git-extended.git
   * [new tag]         v1.0.0 -> v1.0.0
[visimonova@visimonova git-extended]$ gh release create v1.0.0 -F CHANGELOG.md
https://github.com/VISimonova/git-extended/releases/tag/v1.0.0
```

Рис. 4.19: Отправка файлов

Проверяю созданный релиз вводя полученную ссылку в адресную строку (рис. 4.20).

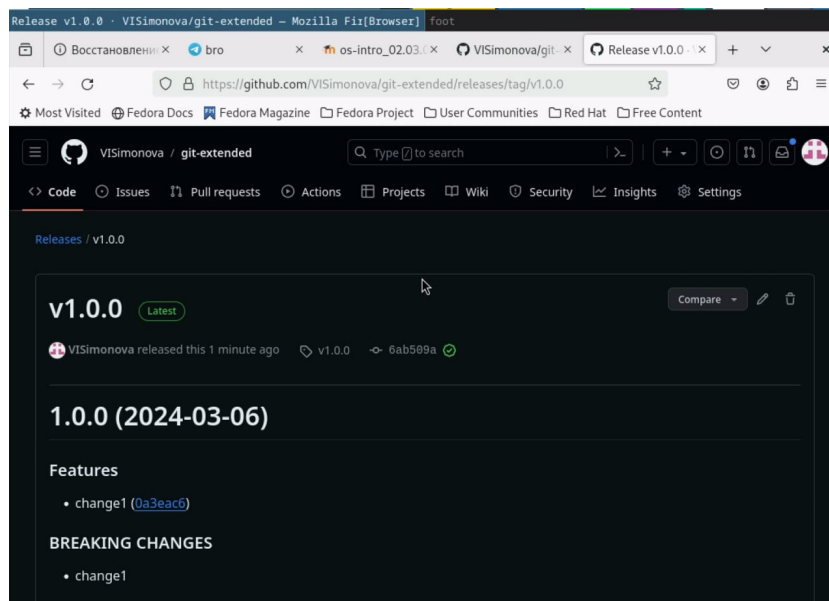


Рис. 4.20: Готовый релиз

Создаю ветку для новой функциональности и сразу же объединяю её с веткой разработки (рис. 4.21).

```
[visimonova@visimonova 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

[visimonova@visimonova 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 e18f26a).

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'
```

Рис. 4.21: Новая ветка

Создаю релиз с версией 1.2.3 (рис. 4.22).

```
[visimonova@visimonova git-extended]$ git flow release start 1.2.3
Переключились на новую ветку «release/1.2.3»

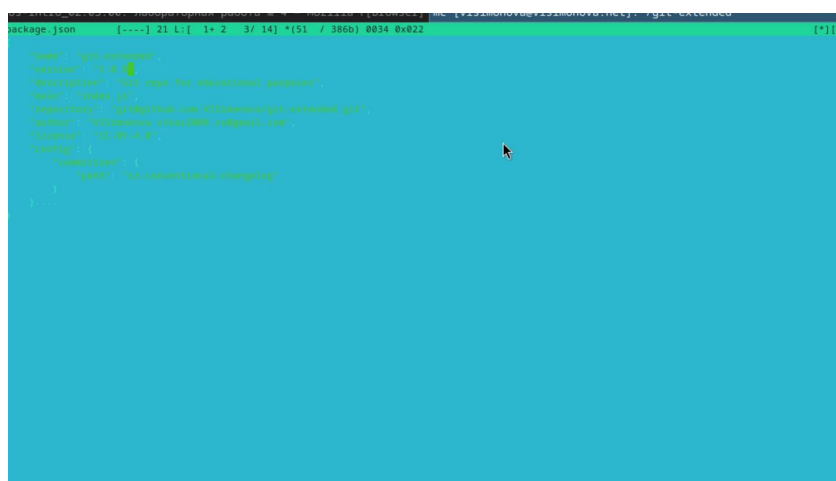
Summary of actions:
- A new branch 'release/1.2.3' was created, based on 'develop'
- You are now on branch 'release/1.2.3'

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'
```

Рис. 4.22: Новый релиз

Изменяю номер версии в файле package.json. (рис. 4.23).



```
package.json [-----] 21 L: [ 1+ 2 3/ 14] * (51 / 386b) 0034 0x022 [*]10
{
  "name": "git-extended",
  "version": "1.2.3",
  "description": "Git tool for educational purposes",
  "main": "index.js",
  "repository": "git@github.com:visimonova/git-extended.git",
  "author": "visimonova <visimonova@yandex.ru>",
  "license": "CC BY-SA 4.0",
  "scripts": {
    "commitizen": "cz",
    "path": "cz-conventional-changelog"
  }
}
```

Рис. 4.23: Новый номер релиза

Создаю журнал изменений. Добавляю журнал изменений в индекс. Отправляю данные (рис. 4.24).


```
[visimonova@visimonova git-extended]$ standard-changelog
✓ output changes to CHANGELOG.md
[visimonova@visimonova git-extended]$ git add CHANGELOG.md
[visimonova@visimonova git-extended]$ git commit -am 'chore(site): update changelog'
[release/1.2.3 92980fe] chore(site): update changelog
2 files changed, 5 insertions(+), 1 deletion(-)
[visimonova@visimonova git-extended]$ git flow release finish 1.2.3
Переключились на ветку «main»
Эта ветка соответствует «origin/main».
Merge made by the 'ort' strategy.
CHANGELOG.md | 4 ++++
package.json | 2 +-
2 files changed, 5 insertions(+), 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».
Merge made by the 'ort' strategy.
CHANGELOG.md | 4 ++++
package.json | 2 +-
2 files changed, 5 insertions(+), 1 deletion(-)
Ветка release/1.2.3 удалена (была 92980fe).

Summary of actions:
- Release branch 'release/1.2.3' has been merged into 'main'
- The release was tagged 'v1.2.3'
- Release tag 'v1.2.3' has been back-merged into 'develop'
- Release branch 'release/1.2.3' has been locally deleted
- You are now on branch 'develop'

[visimonova@visimonova git-extended]$ git push --all
Enter passphrase for key '/home/visimonova/.ssh/id_rsa':
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 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To github.com:VISimonova/git-extended.git
e18f26a..93a36f0 develop -> develop
6ab509a..52a76f7 main -> main
```

Рис. 4.24: Новый номер релиза

Отправляю данные на github. Создаю релиз на github с комментарием из журнала изменений (рис. 4.25).

```
[visimonova@visimonova git-extended]$ git push --tags
Enter passphrase for key '/home/visimonova/.ssh/id_rsa':
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 170 байтов | 170.00 КиБ/с, готово.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:VISimonova/git-extended.git
* [new tag] v1.2.3 -> v1.2.3
[visimonova@visimonova git-extended]$ gh release create v1.2.3 -F CHANGELOG.md
https://github.com/VISimonova/git-extended/releases/tag/v1.2.3
```

Рис. 4.25: Новый номер релиза

Проверяю созданный релиз вводя полученную ссылку в адресную строку (рис. 4.26).

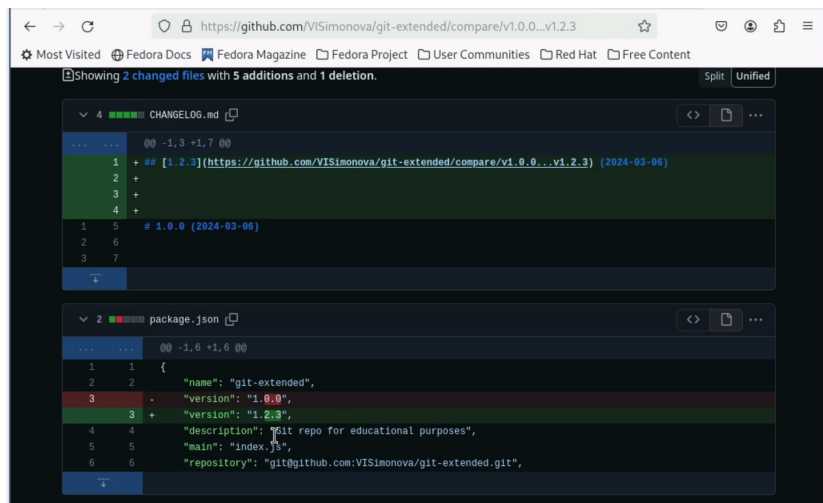


Рис. 4.26: Готовый релиз

5 Выводы

Получила практические навыки работы с репозиториями git

Список литературы

1. Dash, P. Getting Started with Oracle VM VirtualBox / P. Dash. – Packt Publishing Ltd, 2013. – 86 сс.
2. Colvin, H. VirtualBox: An Ultimate Guide Book on Virtualization with VirtualBox. VirtualBox / H. Colvin. – CreateSpace Independent Publishing Platform, 2015. – 70 сс.
3. Vugt, S. van. Red Hat RHCSA/RHCE 7 cert guide : Red Hat Enterprise Linux 7 (EX200 and EX300) : Certification Guide. Red Hat RHCSA/RHCE 7 cert guide / S. van Vugt. – Pearson IT Certification, 2016. – 1008 сс.
4. Робачевский, А. Операционная система UNIX / А. Робачевский, С. Немнюгин, О. Стесик. – 2-е изд. – Санкт-Петербург : БХВ-Петербург, 2010. – 656 сс.
5. Немет, Э. Unix и Linux: руководство системного администратора. Unix и Linux / Э. Немет, Г. Снайдер, Т.Р. Хейн, Б. Уэйли. – 4-е изд. – Вильямс, 2014. – 1312 сс.
6. Колисниченко, Д.Н. Самоучитель системного администратора Linux : Системный администратор / Д.Н. Колисниченко. – Санкт-Петербург : БХВ-Петербург, 2011. – 544 сс.
7. Robbins, A. Bash Pocket Reference / A. Robbins. – O'Reilly Media, 2016. – 156 сс.