

Лабораторная работа №4

Дисциплина: Операционные системы

Бызова Мария Олеговна

Содержание

| | | |
|----------|---------------------------------------|-----------|
| 1 | Цель работы | 5 |
| 2 | Задание | 6 |
| 3 | Выполнение лабораторной работы | 7 |
| 3.1 | Установка git-flow. | 7 |
| 3.2 | Установка Node.js. | 7 |
| 3.3 | Настройка Node.js | 8 |
| 3.4 | Общепринятые коммиты. | 9 |
| 4 | Выводы | 17 |
| | Список литературы | 18 |

Список иллюстраций

| | | |
|------|----------------------------------------------------------------------|----|
| 3.1 | Установка git-flow | 7 |
| 3.2 | Установка git-flow | 7 |
| 3.3 | Установка Node.js | 8 |
| 3.4 | Установка Node.js | 8 |
| 3.5 | Запуск | 8 |
| 3.6 | Выполнение команды | 8 |
| 3.7 | Commitizen | 9 |
| 3.8 | Standard-changelog | 9 |
| 3.9 | Клонирование репозитория | 9 |
| 3.10 | Первый коммит и загрузка на GitHub | 10 |
| 3.11 | Конфигурация для пакетов Node.js | 10 |
| 3.12 | Заполнение параметров пакета | 10 |
| 3.13 | Добавление новых файлов и выполнение коммита | 11 |
| 3.14 | Отправка на github | 11 |
| 3.15 | Инициализация git-flow | 11 |
| 3.16 | Проверка ветки | 11 |
| 3.17 | Загрузка репозитория в хранилище | 12 |
| 3.18 | Установка ветки | 12 |
| 3.19 | Создание релиза | 12 |
| 3.20 | Создание журнала изменений | 12 |
| 3.21 | Добавление журнала изменений в индекс | 13 |
| 3.22 | Заливание релизной ветки в основную ветку | 13 |
| 3.23 | Отправка данных на github | 13 |
| 3.24 | Создание релиза на github | 14 |
| 3.25 | Создание ветки для новой функциональности | 14 |
| 3.26 | Объединение веток | 14 |
| 3.27 | Создание релиза | 15 |
| 3.28 | Обновление версии | 15 |
| 3.29 | Создание журнала изменений | 15 |
| 3.30 | Добавление журнала изменений в индекс | 15 |
| 3.31 | Заливание релизной ветки в основную ветку | 16 |
| 3.32 | Отправка данных на github | 16 |
| 3.33 | Создание релиза на github с комментарием журнала изменений | 16 |

Список таблиц

1 Цель работы

Целью данной лабораторной работы является получение навыков правильной работы с репозиториями git.

2 Задание

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

3 Выполнение лабораторной работы

3.1 Установка git-flow.

Установим git-flow из коллекции репозитория Copr (<https://copr.fedorainfracloud.org/coprs/elegos/gitflow/>) (рис. 3.1 и 3.2).

```
mobihzova@mobihzova ~]$ sudo -i
[sudo] пароль для mobihzova:
[root@mobihzova ~]# 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 поддерживаются 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
Репозиторий успешно подключен.
[root@mobihzova ~]#
```

Рис. 3.1: Установка git-flow

```
copr repo for gitflow owned by elegos
Fedora 39 - x86_64 - Updates
Fedora 39 - x86_64 - Updates
49% [=====]
1.6 kB/s | 1.5 kB 00:00
19 kB/s | 20 kB 00:01
66 kB/s | 108 kB 00:01 ETA
```

Рис. 3.2: Установка git-flow

3.2 Установка Node.js.

На Node.js базируется программное обеспечение для семантического версионирования и общепринятых коммитов. Установим его (рис. 3.3 и 3.4).

```
[root@mobihzova ~]# dnf install nodejs
Последняя проверка окончания срока действия метаданных: 0:00:41 назад, Вт 27 фев 2024 18:49:07.
Зависимости разрешены.
=====
Пакет                Архитектура  Версия                Репозиторий          Размер
-----
Установка:
nodejs                x86_64       1:20.10.0-3.fc39     updates              48 k
Установка зависимостей:
nodejs-libs           x86_64       1:20.10.0-3.fc39     updates              15 M
Установка слабых зависимостей:
nodejs-docs           ppc64le     1:20.10.0-3.fc39     updates              8.1 M
nodejs-full-libs      x86_64       1:20.10.0-3.fc39     updates              8.5 M
nodejs-pnpm           x86_64       1:10.2.3-1.20.10.0.3.fc39 updates              2.2 M
=====
Результат транзакции
Установка 5 Пакетов
```

Рис. 3.3: Установка Node.js

```
foot
[mobihzova@mobihzova ~]$ sudo dnf instal pnpm
```

Рис. 3.4: Установка Node.js

3.3 Настройка Node.js

Теперь запустим (рис. 3.1)

```
[root@mobihzova ~]# 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@mobihzova ~]#
```

Рис. 3.5: Запуск

Выполним следующую команду (рис. 3.6)

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

Рис. 3.6: Выполнение команды

3.4 Общепринятые коммиты.

Введём команду `pnpm add -g commitizen`. Данная программа используется для помощи в форматировании коммитов. При этом устанавливается скрипт `git-cz`, который мы и будем использовать для коммитов (рис. 3.7)

```
no changes to the environment were made. Everything is already up to date.
[root@mobihzova ~]# pnpm add -g commitizen

Update available! 8.12.0 -> 8.15.4.
Changelog: https://github.com/pnpm/pnpm/releases/tag/v8.15.4
Run "pnpm add -g pnpm" to update.

Follow @pnpmjs for updates: https://twitter.com/pnpmjs

Packages: +152
Progress: resolved 152, reused 0, downloaded 152, added 152, done
/root/.local/share/pnpm/global/5:
+ commitizen 4.3.0
Done in 5.9s
[root@mobihzova ~]#
```

Рис. 3.7: Commitizen

Введём команду `pnpm add -g standard-changelog`. Данная программа используется для помощи в создании логов. (рис. 3.8)

```
Done in 3.9s
[root@mobihzova ~]# 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 3.2s
[root@mobihzova ~]#
```

Рис. 3.8: Standard-changelog

Теперь создадим репозиторий `git` и подключим репозиторий к `github`. Предварительно создаём репозиторий на `GitHub` и клонируем его к себе (рис. 3.9)

```
[mobihzova@mobihzova ~]$ git clone --recursive https://github.com/mobyzoa/git-extended.git
Клонирование в «git-extended»...
warning: Похоже, что вы клонировали пустой репозиторий.
```

Рис. 3.9: Клонирование репозитория

Сделаем первый коммит и выложим на `GitHub` (рис. 3.10)

```

mobihzova@mobihzova ~]$ cd git-extended/
mobihzova@mobihzova git-extended]$ git add .
mobihzova@mobihzova git-extended]$ git commit -m "first commit"
main (корневой коммит) f2a00a3] first commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 README.md
mobihzova@mobihzova git-extended]$ git remote add origin https://github.com/mobyzova/git-extended.git
error: внешний репозиторий origin уже существует
mobihzova@mobihzova git-extended]$ git push -u origin main
перечисление объектов: 3, готово.
подсчет объектов: 100% (3/3), 865 байтов | 865.00 КиБ/с, готово.
запись объектов: 100% (3/3), 865 байтов | 865.00 КиБ/с, готово.
всего 3 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
to https://github.com/mobyzova/git-extended.git
* [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
mobihzova@mobihzova git-extended]$

```

Рис. 3.10: Первый коммит и загрузка на GitHub

Выполним конфигурацию для пакетов Node.js (рис. 3.11)

```

mobihzova@mobihzova git-extended]$ pnpm init
wrote to /home/mobihzova/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"
}
mobihzova@mobihzova git-extended]$

```

Рис. 3.11: Конфигурация для пакетов Node.js

Теперь необходимо заполнить несколько параметров пакета (рис. 3.12)

```

mobihzova - Thunar
GNU nano 7.2 package.json
"name": "git-extended",
"version": "1.0.0",
"description": "Git repo for educational purposes",
"main": "index.js",
"keywords": [],
"repository": "https://github.com/mobyzova/git-extended.git",
"author": "Maria Byzova 1132236129@pfur.ru",
"license": "CC-BY-4.0"
"config": {
  "commitizen": {
    "path": "cz-conventional-changelog"
  }
}
}

```

Рис. 3.12: Заполнение параметров пакета

Добавим новые файлы и выполним коммит (рис. 3.13)

```
mobihzova@mobihzova git-extended]$ git add .
mobihzova@mobihzova git-extended]$ git cz
cz-cl1@4.3.0, cz-conventional-changelog@3.3.0

Select the type of change that you're committing: (Use arrow keys)
> feat: A new feature
fix: A bug fix
docs: Documentation only changes
style: Changes that do not affect the meaning of the code (white-space, formatting, missing semi-colons, etc)
refactor: A code change that neither fixes a bug nor adds a feature
perf: A code change that improves performance
test: Adding missing tests or correcting existing tests
Move up and down to reveal more choices)
```

Рис. 3.13: Добавление новых файлов и выполнение коммита

Отправим на github (рис. 3.14)

```
create mode 100644 package.json
[mobihzova@mobihzova git-extended]$ git push
Перечисление объектов: 4, готово.
Подсчет объектов: 100% (4/4), готово.
При сжатии изменений используется до 2 потоков
Сжатие объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 1.16 Киб | 1.16 Миб/с, готово.
Всего 3 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
To https://github.com/mobihzova/git-extended.git
 f2a00a3..170a17b  main -> main
[mobihzova@mobihzova git-extended]$
```

Рис. 3.14: Отправка на github

Инициализируем git-flow и префикс для ярлыков установим в v (рис. 3.15)

```
[mobihzova@mobihzova git-extended]$ git flow init -f
Which branch should be used for bringing forth production releases?
- develop
- main
Branch name for production releases: [main]

Which branch should be used for integration of the "next release"?
- develop
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/mobihzova/git-extended/.git/hooks]
[mobihzova@mobihzova git-extended]$
```

Рис. 3.15: Инициализация git-flow

Проверим, что мы на ветке develop (рис. 3.16)

```
[mobihzova@mobihzova git-extended]$ git branch
* develop
main
[mobihzova@mobihzova git-extended]$
```

Рис. 3.16: Проверка ветки

Загрузим весь репозиторий в хранилище (рис. 3.17)

```
main
[mobihzova@mobihzova git-extended]$ git push --all
Всего 0 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:   https://github.com/mobyzova/git-extended/pull/new/develop
remote:
To https://github.com/mobyzova/git-extended.git
 * [new branch]      develop -> develop
[mobihzova@mobihzova git-extended]$
```

Рис. 3.17: Загрузка репозитория в хранилище

Установим внешнюю ветку как вышестоящую для этой ветки (рис. 3.18)

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

Рис. 3.18: Установка ветки

Создадим релиз с версией 1.0.0 (рис. 3.19)

```
[mobihzova@mobihzova 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'

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

Рис. 3.19: Создание релиза

Создадим журнал изменений (рис. 3.20)

```
[mobihzova@mobihzova git-extended]$ standard-changelog --first-release
✓ created CHANGELOG.md
✓ output changes to CHANGELOG.md
[mobihzova@mobihzova git-extended]$
```

Рис. 3.20: Создание журнала изменений

Добавим журнал изменений в индекс (рис. 3.21)

```
[mobihzova@mobihzova git-extended]$ git add CHANGELOG.md
[mobihzova@mobihzova git-extended]$ git commit -am 'chore(site): add changelog'
[release/1.0.0 41c2c1b] chore(site): add changelog
1 file changed, 9 insertions(+)
create mode 100644 CHANGELOG.md
[mobihzova@mobihzova git-extended]$
```

Рис. 3.21: Добавление журнала изменений в индекс

Зальём релизную ветку в основную ветку (рис. 3.22)

```
create mode 100644 CHANGELOG.md
[mobihzova@mobihzova 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 | 9 ++++++++
1 file changed, 9 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)
fatal: нет описания метки?
fatal: Tagging failed. Please run finish again to retry.
[mobihzova@mobihzova git-extended]$ git flow release finish 1.0.0
Branches 'main' and 'origin/main' have diverged.
And local branch 'main' is ahead of 'origin/main'.
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 | 9 ++++++++
1 file changed, 9 insertions(+)
create mode 100644 CHANGELOG.md
Ветка release/1.0.0 удалена (была 41c2c1b).

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'

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

Рис. 3.22: Заливание релизной ветки в основную ветку

Отправим данные на github (рис. 3.23)

```
[mobihzova@mobihzova git-extended]$ git push --all
Перечисление объектов: 6, готово.
Подсчет объектов: 100% (6/6), готово.
При сжатии изменений используется до 2 потоков
Сжатие объектов: 100% (5/5), готово.
Запись объектов: 100% (5/5), 2.79 КиБ | 2.79 МиБ/с, готово.
Всего 5 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
To https://github.com/mobyzova/git-extended.git
 170a17b..e655573 develop -> develop
 170a17b..84d32cd main -> main
[mobihzova@mobihzova 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.
Всего 1 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
To https://github.com/mobyzova/git-extended.git
 * [new tag] v1.0.0 -> v1.0.0
[mobihzova@mobihzova git-extended]$
```

Рис. 3.23: Отправка данных на github

Создадим релиз на github. Для этого будем использовать утилиты работы с

github (рис. 3.24)

```
[mobihzova@mobihzova git-extended]$ gh release create v1.0.0 -F CHANGELOG.md
https://github.com/mobyzoa/git-extended/releases/tag/v1.0.0
[mobihzova@mobihzova git-extended]$
```

Рис. 3.24: Создание релиза на github

Создадим ветку для новой функциональности (рис. 3.25)

```
[mobihzova@mobihzova 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

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

Рис. 3.25: Создание ветки для новой функциональности

По окончании разработки новой функциональности следующим шагом следует объединить ветку feature_branch с develop (рис. 3.26)

```
[mobihzova@mobihzova 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 e655573).

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'

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

Рис. 3.26: Объединение веток

Создадим релиз с версией 1.2.3 (рис. 3.27)

```
[mobihzova@mobihzova 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'

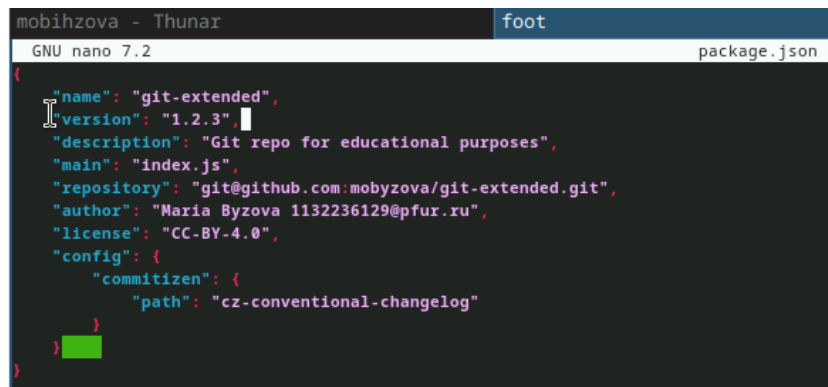
Follow-up actions:
- Bump the version number now!
- Start committing last-minute fixes in preparing your release
- When done, run:

    git flow release finish '1.2.3'

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

Рис. 3.27: Создание релиза

Обновим номер версии в файле package.json (рис. 3.28)



```
mobihzova - Thunar
GNU nano 7.2 package.json
{
  "name": "git-extended",
  "version": "1.2.3",
  "description": "Git repo for educational purposes",
  "main": "index.js",
  "repository": "git@github.com:mobihzova/git-extended.git",
  "author": "Maria Byzova 1132236129@pfur.ru",
  "license": "CC-BY-4.0",
  "config": {
    "commitizen": {
      "path": "cz-conventional-changelog"
    }
  }
}
```

Рис. 3.28: Обновление версии

Создадим журнал изменений (рис. 3.29)

```
[mobihzova@mobihzova git-extended]$ nano package.json
[mobihzova@mobihzova git-extended]$ standard-changelog
✓ output changes to CHANGELOG.md
[mobihzova@mobihzova git-extended]$
```

Рис. 3.29: Создание журнала изменений

Добавим журнал изменений в индекс (рис. 3.30)

```
[mobihzova@mobihzova git-extended]$ git add CHANGELOG.md
[mobihzova@mobihzova git-extended]$ git commit -am 'chore(site): update changelog'
[release/1.2.3 d69494c] chore(site): update changelog
2 files changed, 5 insertions(+), 1 deletion(-)
[mobihzova@mobihzova git-extended]$
```

Рис. 3.30: Добавление журнала изменений в индекс

Зальём релизную ветку в основную ветку (рис. 3.31)

```
[mobihzova@mobihzova git-extended]$ git add CHANGELOG.md
[mobihzova@mobihzova git-extended]$ git commit -am 'chore(site): update changelog'
[release/1.2.3 d69494c] chore(site): update changelog
2 files changed, 5 insertions(+), 1 deletion(-)
[mobihzova@mobihzova 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 d69494c).

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'

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

Рис. 3.31: Заливание релизной ветки в основную ветку

Отправим данные на github (рис. 3.32)

```
[mobihzova@mobihzova git-extended]$ git push --all
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
При сжатии изменений используется до 2 потоков
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 2.77 КБ | 2.77 МБ/с, готово.
Всего 6 (изменений 2), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/mobyzova/git-extended.git
 e655573..242bca1 develop -> develop
 84d32cd..4b5a46a main -> main
[mobihzova@mobihzova git-extended]$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 151 байт | 151.00 КБ/с, готово.
Всего 1 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
To https://github.com/mobyzova/git-extended.git
 * [new tag] v1.2.3 -> v1.2.3
[mobihzova@mobihzova git-extended]$
```

Рис. 3.32: Отправка данных на github

Создадим релиз на github с комментарием из журнала изменений (рис. 3.33)

```
[new tag] v1.2.3 -> v1.2.3
[mobihzova@mobihzova git-extended]$ gh release create v1.2.3 -F CHANGELOG.md
https://github.com/mobyzova/git-extended/releases/tag/v1.2.3
[mobihzova@mobihzova git-extended]$
```

Рис. 3.33: Создание релиза на github с комментарием журнала изменений

4 Выводы

В ходе выполнения данной лабораторной работы я получила навыки правильной работы с репозиториями git.

Список литературы

1. Лабораторная работа №4 [Электронный ресурс] URL: <https://esystem.rudn.ru/mod/page/view.php?id=11111>