

Лабораторная работа №4

Продвинутое использование git

Дагделен З. Р.

9 марта 2024

Российский университет дружбы народов, Москва, Россия

Вводная часть

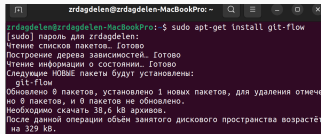
Получение навыков правильной работы с репозиториями git.

1. Установка программного обеспечения

- Установка git-flow
- Установка Node.js
- Настройка Node.js
- Общепринятые коммиты

Выполнение лабораторной работы.

Так как у меня Ubuntu, то команды различаются. Установлю git-flow (рис. 1).



```
zrdagdelen@zrdagdelen-MacBookPro: ~  
zrdagdelen@zrdagdelen-MacBookPro:~$ sudo apt-get install git-flow  
[sudo] пароль для zrdagdelen:  
Чтение списков пакетов... Готово  
Построение дерева зависимостей... Готово  
Чтение информации о состоянии... Готово  
Следующие НОВЫЕ пакеты будут установлены:  
  git-flow  
Обновлено 0 пакетов, установлено 1 новых пакетов, для удаления отмечено 0 пакетов, и 0 пакетов не обновлено.  
Необходимо скачать 38,6 kB архивов.  
После данной операции объем занятого дискового пространства возрастёт на 329 kB.
```

Рис. 1: Установка git-flow через терминал Ubuntu

Обновляю список пакетов и устанавливаю Node.js (рис. 2)

```
zrdagdelen@zrdagdelen-MacBookPro:~$ sudo apt update
Сущ:1 http://ru.archive.ubuntu.com/ubuntu jammy InRelease
Сущ:2 http://ru.archive.ubuntu.com/ubuntu jammy-updates InRelease
Сущ:3 http://ru.archive.ubuntu.com/ubuntu jammy-backports InRelease
Сущ:4 http://security.ubuntu.com/ubuntu jammy-security InRelease
Сущ:5 https://ppa.launchpadcontent.net/obsproject/obs-studio/ubuntu
jammy InRelease
Чтение списков пакетов... готово
Построение дерева зависимостей... готово
Чтение информации о состоянии... готово
Все пакеты имеют последние версии.
zrdagdelen@zrdagdelen-MacBookPro:~$ sudo apt install nodejs
Чтение списков пакетов... готово
Построение дерева зависимостей... готово
Чтение информации о состоянии... готово
Будут установлены следующие дополнительные пакеты:
libc-ares2 libjs-highlight.js libnode72 nodejs-doc
Предлагаемые пакеты:
  npm
Следующие НОВЫЕ пакеты будут установлены:
libc-ares2 libjs-highlight.js libnode72 nodejs nodejs-doc
```

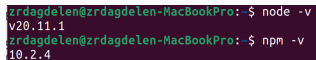
Рис. 2: Обновление пакетов и установка nodejs через терминал Ubuntu

Устанавливаю npm(рис. 3)

```
zrdagdelen@zrdagdelen-MacBookPro: ~$ sudo apt install npm
Чтение списков пакетов... Готово
Построение дерева зависимостей... Готово
Чтение информации о состоянии... Готово
Будут установлены следующие дополнительные пакеты:
gyp libjs-events libjs-inherits libjs-is-typedarray libjs-psl
libjs-source-map libjs-sprintf-js libjs-typedarray-to-buffer
libnode-dev libssl-dev libuv1-dev node-abab node-abbrev
node-agent-base node-ansi-regex node-ansi-styles
node-ansistyles node-aproba node-archy node-are-we-there-yet
node-argparse node-arrify node-asap node-asynckit
node-balanced-match node-brace-expansion node-bulldoze
node-cacache node-chalk node-chownr node-clean-yaml-object
node-cli-table node-clone node-color-convert node-color-name
node-colors node-columnify node-combined-stream node-commander
node-console-control-strings node-copy-concurrently
node-core-util-is node-coveralls node-csson node-cssstyle
node-debug node-decompress-response node-defaults
node-delayed-stream node-delegates node-depd node-diff
node-encoding node-end-of-stream node-err-code
node-escape-string-regexp node-esprima node-events
node-fancy-log node-fetch node-foreground-child node-form-data
```

Рис. 3: Установка npm через терминал Ubuntu

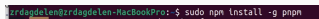
проверяю установку (рис. 4).

A terminal window with a dark background and light green text. The prompt is 'zrdagdelen@zrdagdelen-MacBookPro:~\$'. The first command is 'node -v' and the output is 'v20.11.1'. The second command is 'npm -v' and the output is '10.2.4'.

```
zrdagdelen@zrdagdelen-MacBookPro:~$ node -v
v20.11.1
zrdagdelen@zrdagdelen-MacBookPro:~$ npm -v
10.2.4
```

Рис. 4: Проверка

Устанавливаю PNPM глобально на своей системе (рис. 5).

A terminal window with a dark background. The prompt is 'zrdagdelen@zrdagdelen-MacBookPro:'. The command entered is '\$ sudo npm install -g pnpm'. The output is 'npm WARN config shell "/usr/bin/fish"' and 'npm WARN deprecated rimraf@2.7.1: Use rimraf-version-3 instead, moved to https://www.npmjs.com/package/rimraf-v3-transmuted'.

```
zrdagdelen@zrdagdelen-MacBookPro:~$ sudo npm install -g pnpm
npm WARN config shell "/usr/bin/fish"
npm WARN deprecated rimraf@2.7.1: Use rimraf-version-3 instead, moved to https://www.npmjs.com/package/rimraf-v3-transmuted
```

Рис. 5: Установка pnpm через терминал Ubuntu

Установка программного обеспечения. Настройка Node.js

Для работы с Node.js добавляю каталог с исполняемыми файлами, устанавливаемыми yarn, в переменную PATH (рис. 6). Запускаю `pnpm setup` и выполняю `source ~/.bashrc` (рис. 7).

```
zrdagdelen@zrdagdelen-MacBookPro:~$ export PNPM_HOME="/home/zrdagdelen/.local/share/pnpm"
zrdagdelen@zrdagdelen-MacBookPro:~$ export PATH="$PNPM_HOME:$PATH"
zrdagdelen@zrdagdelen-MacBookPro:~$ echo $PATH
/home/zrdagdelen/.local/share/pnpm:/home/zrdagdelen/.local/share/pnpm:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
zrdagdelen@zrdagdelen-MacBookPro:~$
```

Рис. 6: Отправка в PATH

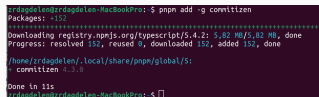
```
zrdagdelen@zrdagdelen-MacBookPro:~$ pnpm setup
Appended new lines to /home/zrdagdelen/.bashrc

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

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

Рис. 7: Запуск setup и выполнение команд

Выполняю команду `pnpm add -g commitizen` (рис. 8).

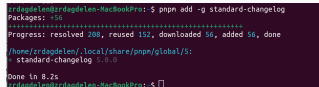


```
zrdagdelen@zrdagdelen-MacBookPro: $ pnpm add -g commitizen
Packages: +152
*****
Downloading registry.npmjs.org/typescript/5.4.2: 5.82 MB/5.82 MB, done
Progress: resolved 152, reused 0, downloaded 152, added 152, done

/home/zrdagdelen/.local/share/pnpm/global/5:
+ commitizen 4.3.0
Done in 11s
zrdagdelen@zrdagdelen-MacBookPro: $
```

Рис. 8: Выполнение команды

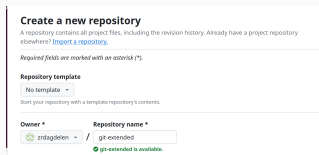
Выполняю `pnpm add -g standard-changelog` (рис. 9).



```
zrdagdelen@zrdagdelen-MacBookPro: $ pnpm add -g standard-changelog
Packages: +56
Progress: resolved 208, reused 152, downloaded 56, added 56, done
/home/zrdagdelen/.local/share/pnpm/global/5:
+ standard-changelog 3.0.0
Done in 8.2s
zrdagdelen@zrdagdelen-MacBookPro: $
```

Рис. 9: Выполнение команды

Создаю репозиторий на GitHub. Назову его git-extended (рис. 10).



The screenshot shows the GitHub 'Create a new repository' page. At the top, it says 'Create a new repository' and provides a brief description of what a repository is. Below this, there's a section for 'Repository template' with a dropdown menu set to 'No template'. Further down, the 'Owner' field is set to 'zrdagdelen' and the 'Repository name' field is set to 'git-extended'. A green checkmark and the text 'git-extended is available.' are displayed below the repository name field.

Create a new repository
A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk ().*

Repository template
No template ▾

Start your repository with a template repository's contents.

Owner * **Repository name ***
zrdagdelen / git-extended
✔ git-extended is available.

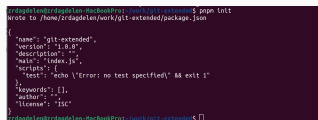
Рис. 10: Создание репозитория на github

Делаю первый коммит и выкладываю на github с помощью последовательности команд (рис. 11).

```
zrdagdelen@zrdagdelen-MacBookPro:~/work/git-extended$ echo "# git-extended" >> README.md
zrdagdelen@zrdagdelen-MacBookPro:~/work/git-extended$ git init
Инициализирован существующий репозиторий Git в /home/zrdagdelen/work/git-extended/.
git/
zrdagdelen@zrdagdelen-MacBookPro:~/work/git-extended$ git add README.md
zrdagdelen@zrdagdelen-MacBookPro:~/work/git-extended$ git commit -m "first commit"
[main 5c3e2a3] first commit
1 file changed, 1 insertion(+)
zrdagdelen@zrdagdelen-MacBookPro:~/work/git-extended$ git branch -M main
zrdagdelen@zrdagdelen-MacBookPro:~/work/git-extended$ git remote add origin git@github.
com:zrdagdelen/git-extended.git
error: внешний репозиторий origin уже существует
zrdagdelen@zrdagdelen-MacBookPro:~/work/git-extended$ git push -u origin main
Пересылание объектов: 0, готово.
Подсчет объектов: 100% (6/6), готово.
При сжатии изменений используется 0% сжатия
Сжатие объектов: 100% (4/4), готово.
Запись объектов: 100% (6/6), 3.11 KiB | 3.11 MiB/s, готово.
Всего 6 (изменения 1), повторно использовано 0 (изменения 0), повторно использовано па
кетов 0
remote: resolving deltas: 100% (1/1), done.
To github.com:zrdagdelen/git-extended.git
 * [new branch]    main -> main
Ветка «main» отслеживает внешнюю ветку «main» из «origin».
zrdagdelen@zrdagdelen-MacBookPro:~/work/git-extended$
```

Рис. 11: Отправка файлов на сервер

Конфигурация для пакетов Node.js с помощью `npm init` (рис. 12).

A terminal window with a dark background and light-colored text. The prompt is 'zrdagden@zrdagden-MacBookPro: ~/work/git-extended\$'. The command 'npm init' has been executed. The output shows the path to the generated package.json file: '/home/zrdagden/work/git-extended/package.json'. Below this, the contents of the package.json file are displayed in a JSON format. The file includes fields for name, version, description, main, scripts, keywords, author, and license.

```
zrdagden@zrdagden-MacBookPro: ~/work/git-extended$ npm init
wrote to /home/zrdagden/work/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"
}

zrdagden@zrdagden-MacBookPro: ~/work/git-extended$
```

Рис. 12: Конфигурация для пакетов Node.js

Меняю файл package.json таким образом, чтобы он приобрел нужный вид. Добавляю новые файлы, выполняю коммит и отправляю на github (рис. 13 - рис. 14).

```
zrdagalemb@zrdagalemb-ncbookPro: ~/work/git-extended$ git add .
zrdagalemb@zrdagalemb-ncbookPro: ~/work/git-extended$ git cz
cz-c(l(84.3.0, cz-conventional-changelog@3.3.0

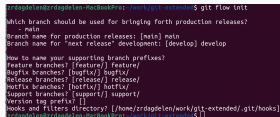
Select the type of change that you're committing: feat:    A new feature
What is the scope of this change (e.g. component or file name): (press enter to skip)
```

Рис. 13: Отправка файлов на GitHub

```
zrdagalemb@zrdagalemb-ncbookPro: ~/work/git-extended$ git push
Пыремчисление объектов: 4, готово.
Пыремчисление объектов: 100% (4/4), готово.
При скачивании изменений используется до 8 потоков.
Скачивание объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 1.14 KiB | 1.14 MiB/s, готово.
Всего 3 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пак
етов 0
to github.com:zrdagalemb/git-extended.git
5c1e3a3..20883ee main -> main
zrdagalemb@zrdagalemb-ncbookPro: ~/work/git-extended$
```

Рис. 14: Отправка файлов на GitHub

Инициализирую git-flow(рис. 15).

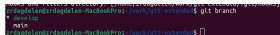


```
zrdagdelen@zrdagdelen-MacBook-Pro: /work/git-extended$ git flow init
Which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main] main
Branch name for "next release" development: [develop] develop

How to name your supporting branch prefixes?
feature branches? [feature/] feature/
bugfix branches? [bugfix/] bugfix/
release branches? [release/] release/
hotfix branches? [hotfix/] hotfix/
support branches? [support/] support/
version tag prefix? []
Hooks and filters directory? [/home/zrdagdelen/work/git-extended/.git/hooks]
zrdagdelen@zrdagdelen-MacBook-Pro: /work/git-extended$
```

Рис. 15: Инициализация git-flow

Проверяю, что нахожусь на ветке develop (рис. 16).



```
zrdagdelen@zrdagdelen-MacBook-Pro: /work/git-extended$ git branch
* develop
main
zrdagdelen@zrdagdelen-MacBook-Pro: /work/git-extended$
```

Рис. 16: Проверка

Загружаю весь репозиторий в хранилище (рис. 17).

```
zrdagdelem@zrdagdelem-HackbookPro: /usr/src/git-extended$ git push --all
Всего 8 (изменений 8), повторно использовано 8 (изменений 8), повторно использовано пак
етов 0
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:   https://github.com/zrdagdelem/git-extended/pull/new/develop
remote:
To github.com:zrdagdelem/git-extended.git
 * [new branch]      develop -> develop
zrdagdelem@zrdagdelem-HackbookPro: /usr/src/git-extended$
```

Рис. 17: Загрузка в хранилище

Устанавливаю внешнюю ветку как вышестоящую для этой ветки и создаю релиз с версией 1.0.0 (рис. 18).

```
zrdagdelem@zrdagdelem-HackbookPro: /usr/src/git-extended$ git branch --set-upstream-to=orig
in/develop develop
Ветка «develop» отслеживает внешнюю ветку «develop» из «origin».
zrdagdelem@zrdagdelem-HackbookPro: /usr/src/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'

zrdagdelem@zrdagdelem-HackbookPro: /usr/src/git-extended$
```

Рис. 18: Настройка веток и создание релиза

-Создаю журнал изменений (standard-changelog --first-release) (рис. 19).

```
iridag@elms:~/iridag/len-NachbookPro:~/work/git-extended$ standard-changelog --first-release
#
✓ created CHANGELOG.md
✓ output changes to CHANGELOG.md
iridag@elms:~/iridag/len-NachbookPro:~/work/git-extended$
```

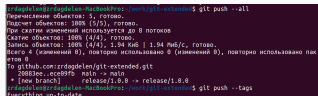
Рис. 19: Создание журнала

Добавляю журнал изменений в индекс и заливаю релизную ветку в основную ветку (рис. 20).

```
iridag@elms:~/iridag/len-NachbookPro:~/work/git-extended$ git add CHANGELOG.md
iridag@elms:~/iridag/len-NachbookPro:~/work/git-extended$ git commit -am 'chore(site): add
changelog'
[release/1.0.0 0304e10] chore(site): add changelog
1 file changed, 9 insertions(+)
create mode 100644 CHANGELOG.md
iridag@elms:~/iridag/len-NachbookPro:~/work/git-extended$ git flow release finish 1.0.0
Переключился на ветку «main»
Диа ветка соответствует «origin/main».
```

Рис. 20: Добавление журнала в индекс и настройка веток

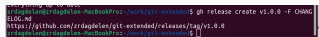
Отправляю данные на github (рис. 21).



```
irradeg@irradeg-laptop:~/work/git-extended$ git push --all
Пересчет объектов: 5, готово.
Пакет объектов: 100% (5/5), готово.
При скатии изменений используется до 8 потоков
Сжатие объектов: 100% (4/4), готово.
Запись объектов: 100% (4/4), 1.94 KiB | 1.94 MiB/s, готово.
Всего 4 (коммитов 0), повторно использовано 0 (изменений 0), повторно использовано пак
етов 0
to github.com:irradeg/git-extended.git
  3683ee... main -> main
+ [new branch] release/1.0.0 -> release/1.0.0
irradeg@irradeg-laptop:~/work/git-extended$ git push --tags
Everything up-to-date
```

Рис. 21: Отправка файлов на GitHub

Создаю релиз на github. Для этого буду использовать утилиты работы с github (рис. 22).



```
zrdagden@zrdagden-MacBookPro2: ~/src/git-extended$ gh release create v1.0.0 -F CHANGELOG.md
https://github.com/zrdagden/git-extended/releases/tag/v1.0.0
zrdagden@zrdagden-MacBookPro2: ~/src/git-extended$
```

Рис. 22: Создание релиза на GitHub

Создаю ветку для новой функциональности (рис. 23).

```
crd@paleo:~/git-extended$ git flow feature start feature_branch
Переходим 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
crd@paleo:~/git-extended$
```

Рис. 23: Создание ветки

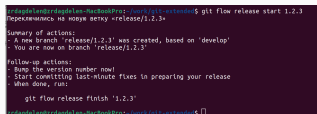
Объединяю ветку feature branch с develop (рис. 24).

```
crd@paleo:~/git-extended$ git flow feature finish feature_branch
Переходим на ветку «develop»
Эта ветка соответствует «origin/develop».
ОК! успешно.
Beta feature/feature_branch уааааа (b7a2a 29883ee).
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'
crd@paleo:~/git-extended$
```

Рис. 24: Объединение веток

Установка программного обеспечения. Общепринятые коммиты

Создаю релиз с версией 1.2.3 (рис. 25).



```
zrdagdelen@zrdagdelen-MacBookPro: ~/work/git-extended$ git flow release start 1.2.3
Переходим 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'

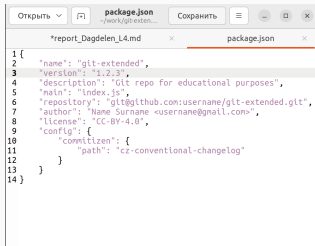
Follow-up actions:
- Bump the version number now!
- Start committing last-minute fixes in preparing your release
- when done, run:

    git flow release finish '1.2.3'

zrdagdelen@zrdagdelen-MacBookPro: ~/work/git-extended$
```

Рис. 25: Создание релиза

Обновляю номер версии в файле package.json. Устанавливаю её в 1.2.3. (рис. 26).

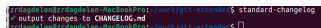


```
package.json
~/work/git-extended...

*report_Dagdelen_L4.md x package.json x

1 {
2   "name": "git-extended",
3   "version": "1.2.3",
4   "description": "Git repo for educational purposes",
5   "main": "index.js",
6   "repository": "git@github.com:username/git-extended.git",
7   "author": "Name Surname <username@gmail.com>",
8   "license": "CC-BY-4.0",
9   "config": {
10     "commitizen": {
11       "path": "cz-conventional-changelog"
12     }
13   }
14 }
```

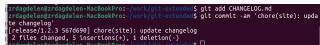

Создаю журнал изменений (рис. 27).



```
zrdagdelanzrdagdelan-MacBookPro: /work/git-extended$ standard-changelog
✓ output changes to CHANGLOG.md
zrdagdelanzrdagdelan-MacBookPro: /work/git-extended$
```

Рис. 27: Создание журнала

Добавляю журнал изменений в индекс (рис. 28).



```
ir@aprilenszraggelen-nickoosProz:~/work/git-extensions$ git add CHANGELOG.md
ir@aprilenszraggelen-nickoosProz:~/work/git-extensions$ git commit -am 'chore(site): update changelog'
[release/1.2.3 567890] chore(site): update changelog
2 files changed, 5 insertions(+), 1 deletion(-)
ir@aprilenszraggelen-nickoosProz:~/work/git-extensions$ git push
Counting objects: 1, done.
Compressing objects: 0%, done.
Writing objects to disk: 0%, done.
To github.com:ir@aprilenszraggelen-nickoosProz:~/work/git-extensions
 * [new branch] 1.2.3 -> 1.2.3

```

Рис. 28: Добавление журнала в индекс

Залью релизную ветку в основную ветку (рис. 29).

```
~/workspace/dagpalm-machookPro: ~$ git flow release finish 1.2.3
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 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 567d69b).
```

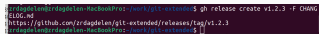
Рис. 29: Настройка веток

Отправляю данные на github (рис. 30).

```
zrddagdel@zrddagdel-MacBookPro: ~/work/git-extended$ git push --all
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
При сжатии изменений используется до 9 потоков.
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 2.78 KiB | 2.78 MiB/c, готово.
Всего 6 (изменений 2), повторно использовано 0 (изменений 0), повторно использовано пак
етов 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To github.com:zrddagdel/git-extended.git
  70891b3..f981879 develop -> develop
  e5c09fb..9888bc3 main -> main
zrddagdel@zrddagdel-MacBookPro: ~/work/git-extended$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 160 байтов | 160.00 KiB/c, готово.
Всего 1 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пак
етов 0
To github.com:zrddagdel/git-extended.git
 * [new tag]         1.2.3 -> 1.2.3
zrddagdel@zrddagdel-MacBookPro: ~/work/git-extended$
```

Рис. 30: Отправка файлов на сервер

Создадим релиз на github с комментарием из журнала изменений (рис. 31).

A terminal window with a dark background and light-colored text. The prompt is 'zrdagden@zrdagden-MacbookPro: ~/work/glt-extended'. The command entered is 'gh release create v1.2.3 -F CHANGELOG.md'. The output shows the GitHub release URL: 'https://github.com/zrdagden/glt-extended/releases/tag/v1.2.3'.

```
zrdagden@zrdagden-MacbookPro: ~/work/glt-extended $ gh release create v1.2.3 -F CHANGELOG.md
https://github.com/zrdagden/glt-extended/releases/tag/v1.2.3
zrdagden@zrdagden-MacbookPro: ~/work/glt-extended $
```

Рис. 31: Создание релиза

Выводы

Я получила навыки правильной работы с репозиториями git.

Список литературы

Операционные системы