

lab 4

Рафи Кази ар

НКАбд-03-24

1032238132@pfur.ru

Лабораторная работа: Преобразование репозитория в Gitflow с использованием Conventional Commits (Ubuntu)

Цель работы

Настроить тестовый Git-репозиторий с использованием **рабочего процесса Gitflow** и **стандартизованных коммитов (Conventional Commits)**, обеспечив структурированную разработку, семантическое версионирование, автоматическое ведение журнала изменений и единый стиль коммитов.

Последовательность выполнения работы

1. Установка необходимого программного обеспечения

1.1 Установка git-flow в Ubuntu git-flow — это расширение для Git, упрощающее использование строгой модели ветвления.

sudo apt update

```
[krafia@pop-os] [07:44 PM] www.krafi.info
└─ * ~ took 3s +2
└─ sudo apt install git-flow
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
git-flow is already the newest version (1.12.3-3).
The following packages were automatically installed and are no longer required:
  asymptote asymptote-doc biber cm-super cm-super-minimal fonts-adf-accanthis fonts-adf-berenis fonts-
  fonts-adf-universalis fonts-arphic-bkai00mp fonts-arphic-bsmi00lp fonts-arphic-gbsn00lp fonts-arphic-
  fonts-cabin fonts-cantarell fonts-comfortaa fonts-dejavu-extra fonts-ebgaramond fonts-ebgaramond-ext
  fonts-gfs-baskerville fonts-gfs-bodoni-classic fonts-gfs-complutum fonts-gfs-didot fonts-gfs-didot-c
  fonts-gfs-neohellenic fonts-gfs-olga fonts-gfs-porson fonts-gfs-solomos fonts-gfs-theokritos fonts-g
  fonts-linuxlibertine fonts-lobster fonts-lobstertwo fonts-oflb-asana-math fonts-open-sans fonts-robo
  fonts-sil-gentium fonts-sil-gentium-basic fonts-sil-gentiumplus fonts-sil-gentiumplus-compact fonts-
  fonts-tlwg-garuda fonts-tlwg-garuda-otf fonts-tlwg-kinnari fonts-tlwg-kinnari-otf fonts-tlwg-laksama
  fonts-tlwg-loma fonts-tlwg-loma-otf fonts-tlwg-mono fonts-tlwg-mono-otf fonts-tlwg-norasi fonts-tlwg-
  fonts-tlwg-purisa-otf fonts-tlwg-sawasdee fonts-tlwg-sawasdee-otf fonts-tlwg-typewriter fonts-tlwg-
  fonts-tlwg-typist-otf fonts-tlwg-typo fonts-tlwg-typo-otf fonts-tlwg-umpush fonts-tlwg-umpush-otf fo
  fonts-tlwg-waree-otf fonts-unfonts-core fonts-unfonts-extra freeglut3 latex-cjk-all latex-cjk-chines
  latex-cjk-chinese-arphic-bkai00mp latex-cjk-chinese-arphic-bsmi00lp latex-cjk-chinese-arphic-gbsn00
  latex-cjk-chinese-arphic-gkai00mp latex-cjk-common latex-cjk-japanese latex-cjk-japanese-wadalab lat
  libautovivification-perl libbtparse2 libbusiness-ismn-perl libbusiness-issn-perl libclass-accessor-p
  libclass-singleton-perl libdata-compare-perl libdata-uniqid-perl libdate-simple-perl libdatetime-cal
  libdatetime-format-builder-perl libdatetime-format-strptime-perl libdatetime-locale-perl libdatetime
  libencode-eucjpm-perl libencode-jis2k-perl libexporter-tiny-perl libfile-find-rule-perl libfile-sha
  libgc1 libglew2.2 libgsl27 libgslcblas0 libipc-run3-perl liblingua-translit-perl liblist-allutils-pe
  liblist-moreutils-xs-perl liblist-someutils-perl liblist-someutils-xs-perl liblist-utilsby-perl libr
  libstyle1c2 libparams-validate-perl libparse-recdescent-perl libperl-io-utf8-strict-perl libregexp-
  libsort-key-perl libtext-bibtex-perl libtext-csv-perl libtext-csv-xs-perl libtext-glob-perl libtext-
  libtext-unidecode-perl libtie-cycle-perl libxml-libxml-perl libxml-libxml-simple-perl libxml-libxslt
  libxml-namespacesupport-perl libxml-sax-base-perl libxml-sax-expat-perl libxml-sax-perl libxml-write
  texinfo texlive-bibtex-extra texlive-fonts-extra texlive-fonts-extra-doc texlive-fonts-extra-links t
  texlive-games texlive-humanities texlive-humanities-doc texlive-lang-arabic texlive-lang-chinese tex
  texlive-lang-cyrillic texlive-lang-czechslovak texlive-lang-english texlive-lang-european texlive-la
  texlive-lang-greek texlive-lang-italian texlive-lang-japanese texlive-lang-korean texlive-lang-other
  texlive-lang-portuguese texlive-lang-spanish texlive-music texlive-publishers texlive-publishers-doc
  texlive-science-doc
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 76 not upgraded.
```

sudo apt install git-flow

□ Пакет git-flow доступен в стандартных репозиториях Ubuntu.

1.2 Установка Node.js и npm Node.js используется для запуска инструментов автоматизации: commitizen, standard-changelog и др.

Установка Node.js sudo apt install nodejs npm

Проверьте установку:

```
[krafia@pop-os] [07:44 PM] www.krafi.info
└─ * ~ took 46s +2
└─ sudo apt install nodejs npm
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nodejs is already the newest version (12.16.1-1~ppc64el).
npm is already the newest version (6.14.11-1~ppc64el).
0 upgraded, 0 newly installed, 0 to remove and 76 not upgraded.
```

node --version

```
[krafi@pop-os ~]$ node --version
v12.22.9
```

npm --version

```
8.5.1
```

Установка pnpm pnpm — быстрый и экономичный пакетный менеджер для Node.js.

```
[krafi@pop-os ~]$ sudo npm install -g pnpm
npm WARN EBADENGINE Unsupported engine {
npm WARN EBADENGINE   package: 'pnpm@10.15.0',
npm WARN EBADENGINE   required: { node: '>=18.12' },
npm WARN EBADENGINE   current: { node: 'v12.22.9',
npm WARN EBADENGINE   }
}

added 1 package, and audited 2 packages in 939ms

1 package is looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

sudo npm install -g pnpm

1.3 Настройка переменной PATH После установки глобальных пакетов pnpm добавляет исполняемые файлы в специальную директорию. Добавим её в PATH.

Выполните команду:

pnpm setup

Система предложит добавить путь в ~/.bashrc. После этого перезагрузите конфигурацию оболочки:

source ~/.bashrc

source ~/.zshrc

Теперь команды, установленные через `pnpm -g`, будут доступны в терминале.

2. Установка инструментов для Conventional Commits

2.1 Установка Commitizen `commitizen` — интерактивная утилита для создания коммитов в формате **Conventional Commits**.

```
[krafi@pop-os ~]$ pnpm add -g commitizen
WARN 2 deprecated subdependencies found: glob@7.2.3, inf
Packages: +151
Progress: resolved 151, reused 0, downloaded 151, added 15
/home/krafi/.local/share/pnpm/global/5:
+ commitizen 4.3.1
Done in 2.5s using pnpm v10.15.0
```

`pnpm add -g commitizen`

После установки становится доступна команда `git cz`.

2.2 Установка standard-changelog Инструмент для автоматического создания и обновления файла `CHANGELOG.md` на основе коммитов.

```
[krafi@pop-os ~]$ pnpm add -g standard-changelog
WARN 2 deprecated subdependencies found: glob@7.2
Packages: +34
Progress: resolved 185, reused 151, downloaded 34,
/home/krafi/.local/share/pnpm/global/5:
+ standard-changelog 7.0.1
Done in 2.2s using pnpm v10.15.0
```

`pnpm add -g standard-changelog`

3. Создание и инициализация Git-репозитория

3.1 Создание репозитория на GitHub

1. Перейдите на <https://github.com>
2. Нажмите **New repository**
3. Назовите репозиторий: `git-extended`
4. Не отмечайте галочки "Add README", ".gitignore", "License" — мы добавим всё вручную

Скопируйте SSH-ссылку (например, `git@github.com:ваше_имя/git-extended.git`)

3.2 Инициализация локального репозитория

```
mkdir git-extended  
cd git-extended  
git init
```

Создадим первый файл:

```
echo "# Git Extended Lab" > README.md
```

Выполним первый коммит:

```
git add .
```

```
[krafi@pop-os] 08:01 PM www.
└─ git-extended [?]
└─ git add .
git commit -m "first commit"
[main d24bdba] first commit
1 file changed, 1 insertion(+)
create mode 100644 README.md

[krafi@pop-os] 08:02 PM www.
└─ git-extended [!]
└─ git push
Username for 'https://gitlab.com': krafi
Password for 'https://krafi@gitlab.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Writing objects: 100% (3/3), 266 bytes | 266.00 KiB/
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://gitlab.com/Krafi.archive/git-extended.git
e3fe2cb..d24bdba main -> main
```

```
git commit -m "first commit"
```

Привяжем к удалённому репозиторию:

```
[Krafi@pop-os] [08:02 PM] www.krafi.info  
└─ git extended took 8s  
→ git checkout -b master  
Switched to a new branch 'master'  
  
[Krafi@pop-os] [08:04 PM] www.krafi.info  
└─ git extended  
→ git push -u origin master  
Username for 'https://gitlab.com': krafi  
Password for 'https://krafi@gitlab.com':  
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0  
remote:  
remote: To create a merge request for master, visit:  
remote: https://gitlab.com/Krafi.archive/git-extended/-/merge_requests/new?merge_request%5Bto_merge_with%5D=main  
remote:  
To https://gitlab.com/Krafi.archive/git-extended.git  
 * [new branch]      master -> master  
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

Замените ваше имя на ваш GitHub-логин.

4.1 Инициализация package.json `pnpm init`

□ Справка о лицензиях:

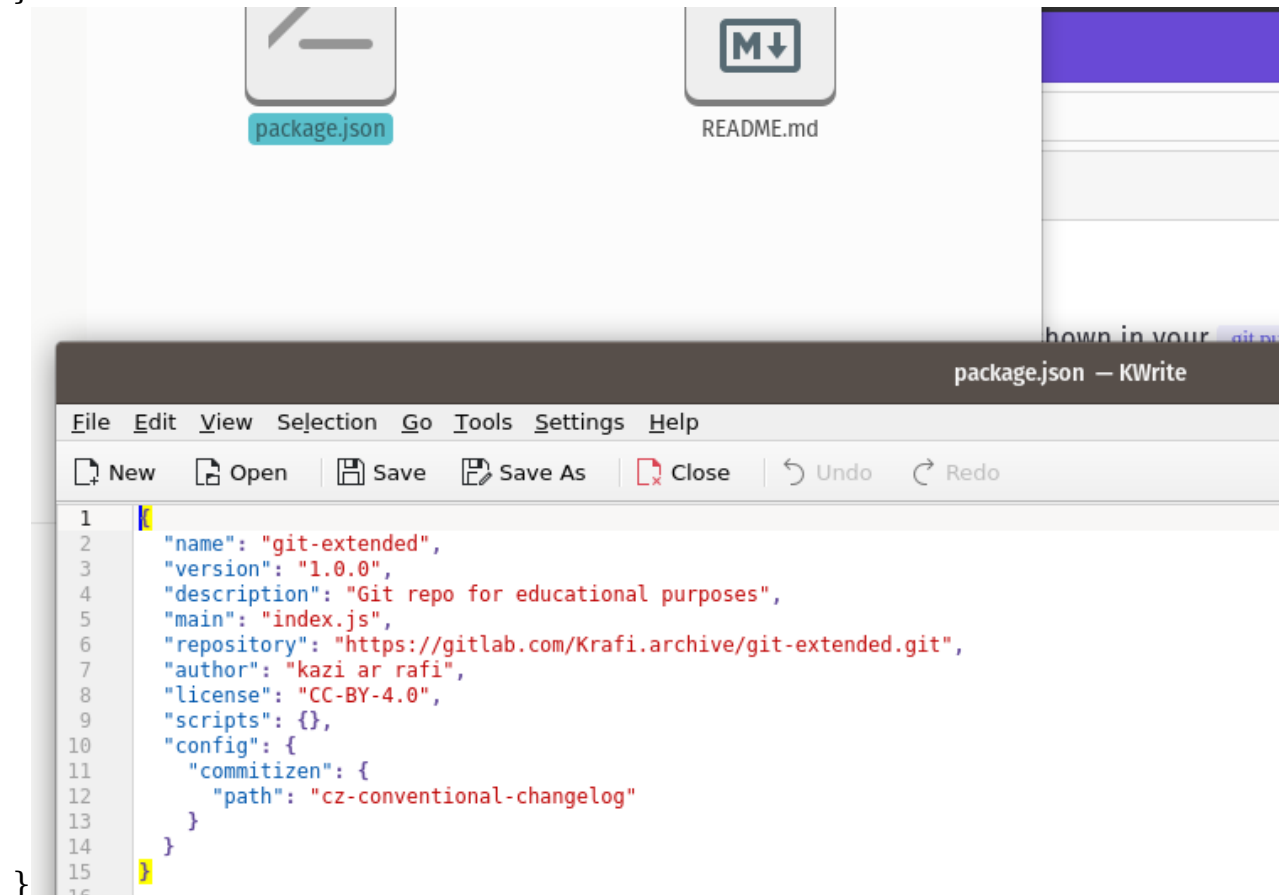
CC-BY-4.0 — это лицензия Creative Commons Attribution 4.0, часто используемая для документации и открытых материалов. Она разрешает свободное использование при указании авторства.

```
{
  "name": "git-extended",
  "version": "1.0.0",
  "description": "Git repo for educational purposes",
  "main": "index.js",
  "repository": "git@github.com:ваше_имя/git-extended.git",
  "author": "Иван Петров <ivan@example.com>",
  "license": "CC-BY-4.0",
}
```

```

"scripts": {},
"config": {
  "commitizen": {
    "path": "cz-conventional-changelog"
  }
}

```



4.3 Фиксация изменений git add package.json

Сделаем коммит с помощью git cz:

git cz

Выберите: - Type: chore - Scope: package - Subject: add commitizen configuration

После завершения:

```

krafi@pop-os ] → [ 08:07 PM ] → www.krafi.info
└─ git-extended is v1.0.0 [?] is v1.0.0 via v18.20.8
→ 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) package - subject: add commitizen configuration
Write a short, imperative tense description of the change (max 44 chars):
(28) add commitizen configuration
Provide a longer description of the change: (press enter to skip)
package - Subject: add commitizen configuration
Are there any breaking changes? No
Does this change affect any open issues? No
[master 5bd5167] chore(package - subject: add commitizen configuration): add commitizen configuration
1 file changed, 15 insertions(+)
create mode 100644 package.json

krafi@pop-os ] → [ 08:10 PM ] → www.krafi.info
└─ git-extended is v1.0.0 [?] is v1.0.0 via v18.20.8 took 2m48s
→ git push
Username for 'https://gitlab.com': krafi
Password for 'https://krafi@gitlab.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 16 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 547 bytes | 547.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: To create a merge request for master, visit:
remote: https://gitlab.com/Krafi.archive/git-extended/-/merge_requests/new?merge_request%5Bsource_branch%5D=master
remote:
To https://gitlab.com/Krafi.archive/git-extended.git
d24bdba..5bd5167 master -> master

```

git push

5. Инициализация Gitflow

Запустите инициализацию:

git flow init

Примите значения по умолчанию, но при запросе префикса для тегов укажите:

Tag prefix? v

После инициализации вы окажетесь на ветке develop.

Проверьте:

git branch

Вывод:

* develop
master

Отправьте все ветки в GitHub:


```

krafi@pop-os ] 08:12 PM > www.krafi.info
└─ git-extended is v1.0.0 is v1.0.0 via v18.20.8 took 16s
git flow init

Which branch should be used for integration of the "next release"?
- main
Branch name for "next release" development: [] main

How to name your supporting branch prefixes?
Feature branches? [feature/] v
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [/home/krafi/Documents/arciture rudn/4/git-extended/git-extended/.git]

krafi@pop-os ] 08:12 PM > www.krafi.info
└─ git-extended is v1.0.0 is v1.0.0 via v18.20.8 took 28s
git branch

krafi@pop-os ] 08:12 PM > www.krafi.info
└─ git-extended is v1.0.0 is v1.0.0 via v18.20.8 took 2s
git push --all
Username for 'https://gitlab.com': krafi
Password for 'https://krafi@gitlab.com':
Everything up-to-date

```

git push --all

Настройте отслеживание для ветки develop:

git branch --set-upstream-to=origin/develop develop

6. Создание первого релиза (v1.0.0)

6.1 Создание ветки релиза git flow release start 1.0.0

6.2 Генерация файла CHANGELOG.md standard-changelog --first-release

Создастся файл CHANGELOG.md со всеми коммитами.

Добавьте его в индекс:

git add CHANGELOG.md

Сделайте коммит:

git commit -am "chore(site): add changelog"

6.3 Завершение релиза git flow release finish 1.0.0

Эта команда: - Сольёт изменения в master - Создаст тег v1.0.0 - Сольёт изменения обратно в develop - Удалит ветку

```

[❏ krafia@pop-os ]—————[❏ 08:15 PM]—————> www.krafi.info
└─❏ git-extended is 📦 v1.0.0 is 📦 v1.0.0 via 🌐v18.20
> git flow release start 1.0.0
Switched to a new branch 'release/1.0.0'

Summary of actions:
- A new branch 'release/1.0.0' was created, based on 'main'
- 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'

[❏ krafia@pop-os ]—————[❏ 08:15 PM]—————> www.krafi.info
└─❏ git-extended
> standard-changelog --first-release

[❏ krafia@pop-os ]—————[❏ 08:15 PM]—————> www.krafi.info
└─❏ git-extended [?]
> git add CHANGELOG.md

[❏ krafia@pop-os ]—————[❏ 08:15 PM]—————> www.krafi.info
└─❏ git-extended [+]
> git commit -am "chore(site): add changelog"
[release/1.0.0 e8a7d97] chore(site): add changelog
1 file changed, 4 insertions(+)
create mode 100644 CHANGELOG.md

[❏ krafia@pop-os ]—————[❏ 08:15 PM]—————> www.krafi.info
└─❏ git-extended
> git flow release finish 1.0.0
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
Merge made by the 'ort' strategy.
CHANGELOG.md | 4 ++++
1 file changed, 4 insertions(+)
create mode 100644 CHANGELOG.md
Already on 'master'
Your branch is ahead of 'origin/master' by 2 commits.
(use "git push" to publish your local commits)
fatal: no tag message?
release/1.0.0Fatal: Tagging failed. Please run finish again to retry.

```

6.4 Отправка в GitHub `git push --all`
`git push --tags`

7. Создание релиза на GitHub

Установите GitHub CLI (если ещё не установлен):

`sudo apt install gh`

Войдите в аккаунт:

```
[krafi@pop-os ] [08:22 PM] www.krafi.info
└─ git-extended is v1.0.0 is v1.0.0 via v18.20.8 took 56s
$ gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations? HTTPS
? Authenticate Git with your GitHub credentials? No
? How would you like to authenticate GitHub CLI? Paste an authentication token
Tip: you can generate a Personal Access Token here https://github.com/settings/tokens
The minimum required scopes are 'repo', 'read:org'.
? Paste your authentication token: *****
- gh config set -h github.com git_protocol https
✓ Configured git protocol
✓ Logged in as View-my-Git-Lab-krafi
```

`gh auth login`

Создайте релиз на GitHub, используя содержимое CHANGELOG.md:

`gh release create v1.0.0 -F CHANGELOG.md`

8. Разработка новой функциональности

8.1 Создание ветки feature `git flow feature start feature_branch`

Создадим пример функции:

```
echo "console.log('Новая функция!');" > feature.js
git add feature.js
```

Сделаем коммит через `git cz`:

```

krafi@pop-os ] 08:23 PM www.krafi.info
└─ git-extended
└─ git flow feature start feature_branch
Fatal: Branch 'vfeature_branch' already exists. Pick another name.

krafi@pop-os ] 08:23 PM www.krafi.info
└─ git-extended
> echo "console.log('Новая функция!');" > feature.js
git add feature.js

krafi@pop-os ] 08:24 PM www.krafi.info
└─ git-extended [+] via v18.20.8
└─ git cz
Aborting commit due to empty commit message.

```

git cz

Выберите: - Type: feat - Scope: core - Subject: add new feature module

Отправим ветку:

git push origin feature/feature_branch

8.2 Завершение разработки Когда функция готова:

git flow feature finish feature_branch

Команда: - Сольёт feature_branch в develop - Удалит ветку - Переключится на develop

Отправьте изменения:

git push

git flow release finish 1.0.0

9. Подготовка нового релиза (v1.2.3)

Предположим, что в develop уже есть новые функции и исправления.

```

[krafi@pop-os ~]$ git flow release start 1.2.3
Switched to a new branch 'release/1.2.3'

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

Follow-up actions:
- Bump the version number in package.json.
- Start committing last-minute changes.
- When done, run: git flow release finish

```

9.1 Создание ветки релиза `git flow release start 1.2.3`

9.2 Обновление версии в `package.json` Откройте `package.json` и измените:

```

GNU nano 6.2 package.json
{
  "name": "git-extended",
  "version": "1.3.2",
  "description": "Git repo for educational purposes",
  "main": "index.js",
  "repository": "https://gitlab.com/Krafi.archive/git-extended.git",
  "author": "kazi ar rafi",
  "license": "CC-BY-4.0",
  "scripts": {},
  "config": {
    "commitizen": {
      "path": "cz-conventional-changelog"
    }
  }
}

```

"version": "1.2.3"

Сохраните и добавьте файл:

```
git add package.json
```

Сделайте коммит:

```
git commit -m "chore(release): bump version to 1.2.3"
```

9.3 Обновление журнала изменений `standard-changelog`

Инструмент добавит новые изменения в `CHANGELOG.md`.

Зафиксируем:

```
git add CHANGELOG.md
```

```
git commit -am "chore(site): update changelog"
```


9.4 Завершение релиза git flow release finish 1.2.3

```

[krafi@pop-os ~]$ git extended i
└─ nano package.json

[krafi@pop-os ~]$ git extended i
└─ git add package.json

[krafi@pop-os ~]$ git extended i
└─ git commit -m "chore(release/1.2.3 a066719) chore
[release/1.2.3 a066719] chore
1 file changed, 1 insertion

[krafi@pop-os ~]$ git extended i
└─ standard-changelog

[krafi@pop-os ~]$ git extended i
└─ git add CHANGELOG.md
git commit -am "chore(site):
[release/1.2.3 91e86f8] chore
1 file changed, 4 insertions

[krafi@pop-os ~]$ git extended i
└─ git flow release finish
Branches 'main' and 'origin/
And local branch 'main' is a
Switched to branch 'master'
Your branch is up to date wi
Merge made by the 'ort' stra
CHANGELOG.md | 4 ++++
package.json | 2 +-
2 files changed, 5 insertio
Already on 'master'
Your branch is ahead of 'ori
(use "git push" to publish
Switched to branch 'main'
Your branch is ahead of 'ori
(use "git push" to publish
Merge made by the 'ort' stra
CHANGELOG.md | 4 ++++
package.json | 2 +-
2 files changed, 5 insertio
Deleted branch release/1.2.3

Summary of actions:
- Release branch 'release/1.
- The release was tagged '1.
- Release tag '1.2.3' has be
- Release branch 'release/1.
- You are now on branch 'mai

```


9.5 Отправка в GitHub `git push --all` `git push --tags`

9.6 Создание релиза на GitHub `gh release create v1.2.3 -F`

```
[~] krafi@pop-os [~] [~] 08:31 PM [~] www.krafi.info
[~] git-extended is v1.3.2 [~] is v1.3.2 via v18.20.8 took 16s
[~] git push --all
git push --tags
Username for 'https://gitlab.com': krafi
Password for 'https://krafi@gitlab.com':
Enumerating objects: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 16 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (9/9), 1.20 KiB | 1.21 MiB/s, done.
Total 9 (delta 3), reused 0 (delta 0), pack-reused 0
remote:
remote: To create a merge request for master, visit:
remote:   https://gitlab.com/Krafi.archive/git-extended/-/merge_requests/new?merge_request%5Bsource_branch%5D=master
remote:
To https://gitlab.com/Krafi.archive/git-extended.git
   d24bdba..6214867 main -> main
   c9c12eb..05c7736 master -> master
Username for 'https://gitlab.com': krafi
Password for 'https://krafi@gitlab.com':
Enumerating objects: 2, done.
Counting objects: 100% (2/2), done.
Delta compression using up to 16 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 286 bytes | 286.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0
To https://gitlab.com/Krafi.archive/git-extended.git
 * [new tag]         1.0.0 -> 1.0.0
 * [new tag]         1.2.3 -> 1.2.3
```

CHANGELOG.md