

# **Лабораторная работа №4**

**Продвинутое использование git**

Карпова Есения Алексеевна

# **Содержание**

<b>1 Цель работы</b>	<b>5</b>
<b>2 Задание</b>	<b>6</b>
<b>3 Теоретическое введение</b>	<b>7</b>
<b>4 Выполнение лабораторной работы</b>	<b>8</b>
<b>5 Выводы</b>	<b>17</b>

# Список иллюстраций

4.1 Установка git-flow . . . . .	8
4.2 Установка Node.js . . . . .	8
4.3 Установка curl . . . . .	9
4.4 HTTP-запрос . . . . .	9
4.5 Ввод команды . . . . .	9
4.6 Commitizen . . . . .	10
4.7 Standard-changelog . . . . .	10
4.8 Создание репозитория git . . . . .	10
4.9 Конфигурация для пакетов Node.js . . . . .	11
4.10 Добавление новых файлов . . . . .	12
4.11 Конфигурация git-flow . . . . .	12
4.12 Проверка нахождения на ветке develop . . . . .	12
4.13 Создание релиза . . . . .	13
4.14 Создание журнала изменений . . . . .	13
4.15 Залив релиза . . . . .	13
4.16 Создание релиза на github . . . . .	14
4.17 Создание ветки . . . . .	14
4.18 Обновление версии в файле . . . . .	15
4.19 Создание журнала и добавление его в индекс . . . . .	15
4.20 Залив релизной ветки . . . . .	15
4.21 Отправка данных на github . . . . .	16

# **Список таблиц**

# **1 Цель работы**

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 Теоретическое введение**

Рабочий процесс Gitflow:

Рабочий процесс Gitflow Workflow. Будем описывать его с использованием пакета git-flow.

Общая информация:

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 сливаются с веткой develop. Когда работа над веткой релиза release завершена, она сливаются 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

## 1. Выполнение работы для тестового репозитория

Для начала нужно установить необходимое ПО, в моем случае для ubuntu. Ввожу нужную команду для установки git-flow (рис. [4.1]).

### Ubuntu

- **Ubuntu 18.04:** git-flow AVH edition is packaged with Ubuntu. You can install the last version of git-flow AVH Edition using:

```
$ sudo apt-get install git-flow
```



Рис. 4.1: Установка git-flow

Ввожу нужную команду для установки Node.js (рис. [4.2]).

Then install Node.js:

```
$ sudo apt install nodejs
```

Copy

Рис. 4.2: Установка Node.js

Также мне понадобится curl для работы с данными сервера (рис. [4.3]).

```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~$ sudo apt install curl
[sudo] password for eakarpova:
Чтение списков пакетов... Готово
Построение дерева зависимостей... Готово
Чтение информации о состоянии... Готово
Следующие НОВЫЕ пакеты будут установлены:
  curl
Обновлено 0 пакетов, установлено 1 новых пакетов, для удаления отмечено 0 пакетов, и 57 пакетов не обновлено.
Необходимо скачать 216 kB архивов.
После данной операции объём занятого дискового пространства возрастёт на 490 kB.
Пол:1 http://ru.archive.ubuntu.com/ubuntu mantic-updates/main amd64 curl amd64 8.2.1-1ubuntu3.2 [216 kB]
Получено 216 kB за 0с (1 015 kB/s)
Выбор ранее не выбранного пакета curl.
(Чтение базы данных ... на данный момент установлено 300472 файла и каталога.)
Подготовка к распаковке .../curl_8.2.1-1ubuntu3.2_amd64.deb ...
Распаковывается curl (8.2.1-1ubuntu3.2) ...
Настраивается пакет curl (8.2.1-1ubuntu3.2) ...
Обрабатываются триггеры для man-db (2.11.2-3) ...
```

Рис. 4.3: Установка curl

Использую curl без аргументов для выполнения HTTP-запроса (рис. [4.4]).

```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~$ curl https://raw.githubusercontent.com/creationix/nvm/master/install.sh | bash
% Total    % Received % Xferd  Average Speed   Time     Time   Current
          Dload  Upload   Total Spent  Left Speed
100 16555  100 16555    0      0  10848      0  0:00:01  0:00:01  --- 10848
=> Downloading nvm from git to '/home/eakarpova/.nvm'
=> Cloning into '/home/eakarpova/.nvm'...
remote: Enumerating objects: 365, done.
remote: Counting objects: 100% (365/365), done.
remote: Compressing objects: 100% (312/312), done.
remote: Total 365 (delta 43), reused 169 (delta 27), pack-reused 0
Receiving objects: 100% (365/365), 365.21 KiB | 91.00 KiB/s, done.
Resolving deltas: 100% (43/43), done.
* (HEAD detached at FETCH_HEAD)
  master
=> Compressing and cleaning up git repository

=> Appending nvm source string to /home/eakarpova/.bashrc
=> Appending bash_completion source string to /home/eakarpova/.bashrc
=> Close and reopen your terminal to start using nvm or run the following to use it now:

export NVM_DIR="$HOME/.nvm"
[ -s "$NVM_DIR/nvm.sh" ] && \. "$NVM_DIR/nvm.sh" # This loads nvm
[ -s "$NVM_DIR/bash_completion" ] && \. "$NVM_DIR/bash_completion" # This loads nvm bash_completion
```

Рис. 4.4: HTTP-запрос

Ввожу команду для перезагрузки (рис. [4.5]).

```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~$ source ~/.bashrc
```

Рис. 4.5: Ввод команды

Для форматирования коммитов и скрипта git-cz использую приведенную ниже программу (рис. [4.6]).

```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ npm add -g commitizen  
added 168 packages in 11s  
27 packages are looking for funding  
  run `npm fund` for details
```

Рис. 4.6: Commitizen

Для создания логов использую приведенную ниже программу (рис. [4.7]).

```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ npm add -g standard-changelog  
added 71 packages in 7s  
16 packages are looking for funding  
  run `npm fund` for details
```

Рис. 4.7: Standard-changelog

Создаю репозиторий git-extended, делаю первый коммит и выкладываю на github(рис. [4.8]).

```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~$ cd Desktop/work  
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work$ git clone --recursive https://github.com/eakarpova  
/git-extended.git  
Cloning into 'git-extended'...  
warning: You appear to have cloned an empty repository.  
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work$ cd git-extended/  
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git commit -m "first commit"  
On branch main  
  
Initial commit  
  
nothing to commit (create/copy files and use "git add" to track)  
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git remote add origin git@github.com:  
<username>/git-extended.git  
bash: username: No such file or directory  
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git remote add origin git@github.com:  
eakarpova/git-extended.git  
error: remote origin already exists.
```

Рис. 4.8: Создание репозитория git

Для конфигурации общепринятых коммитов ввожу команду (рис. [4.9]).

```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ npm init -y
Wrote to /home/eakarpova/Desktop/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"
}
```

Рис. 4.9: Конфигурация для пакетов Node.js

Для того чтобы сконфигурировать формат коммитов добавим в исходный файл команду для формирования коммитов (рис. [??]).

```
={
  "name": "git-extended",
  "version": "1.0.0",
  "description": "Git repo for educational purposes",
  "main": "index.js",
  "scripts": {
    "test": "echo \\"Error: no test specified\\" && exit 1"
  },
  "repository": {
    "type": "git",
    "url": "https://github.com/eakarpova/git-extended.git"
  },
  "keywords": [],
  "author": "Esenia Karpova 1132236008@pfur.ru",
  "license": "CC-BY-4.0",
  "config": {
    "commitizen": {
      "path": "cz-conventional-changelog"
    }
  }
}
```

width=100%}

Добавим новые файлы и выполним коммит, после чего отправим на github(рис. [4.10]).

```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git add .
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/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):
(1) a
? 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
[main 9646267] feat: a
1 file changed, 9 insertions(+), 16 deletions(-)
```

Рис. 4.10: Добавление новых файлов

Инициализируем git-flow (рис. [4.11]).

```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/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/eakarpova/Desktop/work/git-extended/.git/hooks]
```

Рис. 4.11: Конфигурация git-flow

Проверим, что находимся на ветке develop (рис. [4.12]).

```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git branch
* develop
  main
```

Рис. 4.12: Проверка нахождения на ветке develop

Загрузим весь репозиторий в хранилище и установим внешнюю ветку как вышестоящую, после чего создадим релиз с версией 1.0.0. (рис. [4.13]).

```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git flow release start 1.0.0
Switched to a new branch 'release1.0.0'

Summary of actions:
- A new branch 'release1.0.0' was created, based on 'develop'
- You are now on branch 'release1.0.0'

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'
```

Рис. 4.13: Создание релиза

Создаем журнал изменений и добавляем его в индекс (рис. [4.14]).

```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ standard-changelog --first-release
✓ created CHANGELOG.md
✓ output changes to CHANGELOG.md
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git add CHANGELOG.md
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git commit -am 'chore(site): add changelog'
[release1.0.0 5821f5d] chore(site): add changelog
 1 file changed, 9 insertions(+)
 create mode 100644 CHANGELOG.md
```

Рис. 4.14: Создание журнала изменений

Зальем релизную ветку в основную и отправим данные на github (рис. [4.15]).

```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git flow release finish 1.0.0
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
Merge made by the 'ort' strategy.
 CHANGELOG.md | 9 ++++++++
 1 file changed, 9 insertions(+)
 create mode 100644 CHANGELOG.md
Already on 'main'
Your branch is ahead of 'origin/main' by 2 commits.
 (use "git push" to publish your local commits)
fatal: no tag message?
Fatal: Tagging failed. Please run finish again to retry.
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git push --all
Username for 'https://github.com': eakarpova
Password for 'https://eakarpova@github.com':
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 16 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 1.90 KiB | 1.90 MiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/eakarpova/git-extended.git
 9646267..6e7ca8d main -> main
 * [new branch]      release1.0.0 -> release1.0.0
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git push --tags
Username for 'https://github.com': eakarpova
Password for 'https://eakarpova@github.com':
Everything up-to-date
```

Рис. 4.15: Залив релиза

Создадим релиз на github с помощью утилиты (рис. [4.16]).

```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ gh release create v1.0.0 -F CHANGELOG
.md
https://github.com/eakarpova/git-extended/releases/tag/v1.0.0
```

Рис. 4.16: Создание релиза на github

2. Преобразование рабочего репозитория в репозиторий с git-flow и conventional commits.

Создадим ветку для новой функциональности (рис. [4.17]).

```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git flow feature start feature_branch
Switched to a new branch 'featurefeature_branch'

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

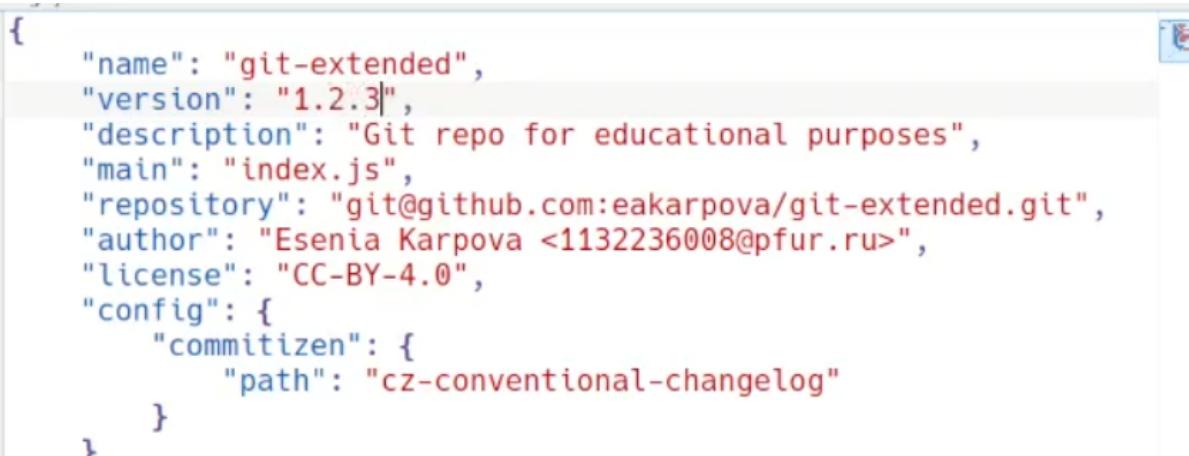
Now, start committing on your feature. When done, use:
  git flow feature finish feature_branch

eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git flow feature finish feature_branch
Switched to branch 'develop'
Already up to date.
Deleted branch featurefeature_branch (was 9646267).

Summary of actions:
- The feature branch 'featurefeature_branch' was merged into 'develop'
- Feature branch 'featurefeature_branch' has been locally deleted
- You are now on branch 'develop'
```

Рис. 4.17: Создание ветки

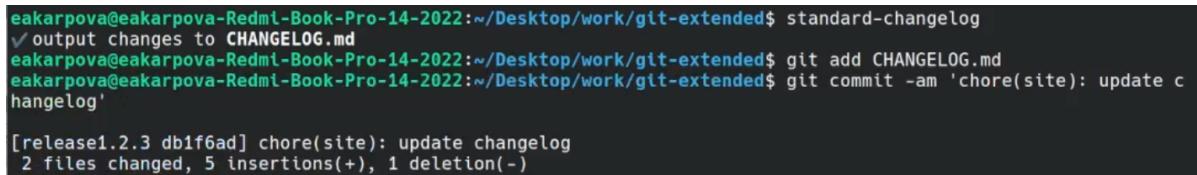
Создадим релиз с версией 1.2.3 и обновим номер версии в файле package.json (рис. [4.18]).



```
{  
  "name": "git-extended",  
  "version": "1.2.3",  
  "description": "Git repo for educational purposes",  
  "main": "index.js",  
  "repository": "git@github.com:eakarpova/git-extended.git",  
  "author": "Esenia Karpova <1132236008@pfur.ru>",  
  "license": "CC-BY-4.0",  
  "config": {  
    "commitizen": {  
      "path": "cz-conventional-changelog"  
    }  
}
```

Рис. 4.18: Обновление версии в файле

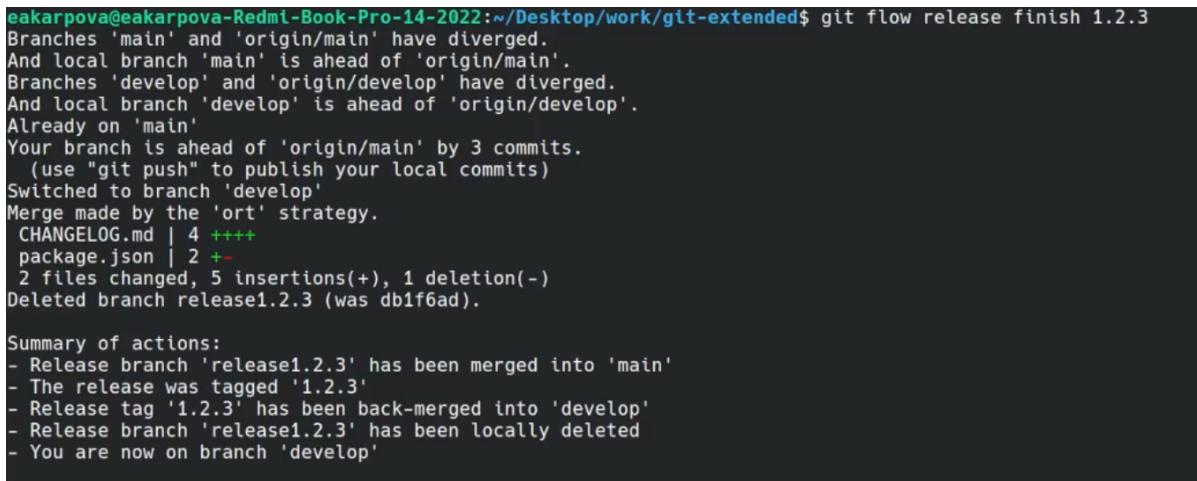
Создадим журнал изменений и добавим его в индекс (рис. [4.19]).



```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ standard-changelog  
✓ output changes to CHANGELOG.md  
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git add CHANGELOG.md  
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git commit -am 'chore(site): update c  
hangelog'  
[release1.2.3 db1f6ad] chore(site): update changelog  
 2 files changed, 5 insertions(+), 1 deletion(-)
```

Рис. 4.19: Создание журнала и добавление его в индекс

Зальем релизную ветку в основную (рис. [4.20]).



```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git flow release finish 1.2.3  
Branches 'main' and 'origin/main' have diverged.  
And local branch 'main' is ahead of 'origin/main'.  
Branches 'develop' and 'origin/develop' have diverged.  
And local branch 'develop' is ahead of 'origin/develop'.  
Already on 'main'  
Your branch is ahead of 'origin/main' by 3 commits.  
(use "git push" to publish your local commits)  
Switched to branch 'develop'  
Merge made by the 'ort' strategy.  
  CHANGELOG.md | 4 +***  
  package.json | 2 +-  
 2 files changed, 5 insertions(+), 1 deletion(-)  
Deleted branch release1.2.3 (was db1f6ad).  
  
Summary of actions:  
- Release branch 'release1.2.3' has been merged into 'main'  
- The release was tagged '1.2.3'  
- Release tag '1.2.3' has been back-merged into 'develop'  
- Release branch 'release1.2.3' has been locally deleted  
- You are now on branch 'develop'
```

Рис. 4.20: Залив релизной ветки

Отправим данные на github (рис. [4.21]).

```
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git push --all
Username for 'https://github.com': eakarpova
Password for 'https://eakarpova@github.com':
Enumerating objects: 12, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 16 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 3.56 KiB | 3.56 MiB/s, done.
Total 7 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/eakarpova/git-extended.git
  9646267..3ee59b1  develop -> develop
    6e7ca8d..208bbff  main -> main
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ git push --tags
Username for 'https://github.com': eakarpova
Password for 'https://eakarpova@github.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), 265 bytes | 265.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/eakarpova/git-extended.git
 * [new tag]      1.0.0 -> 1.0.0
 * [new tag]      1.2.3 -> 1.2.3
eakarpova@eakarpova-Redmi-Book-Pro-14-2022:~/Desktop/work/git-extended$ gh release
.md
https://github.com/eakarpova/releases/tag/v1.2.3
```

Рис. 4.21: Отправка данных на github

(рис. [4.21]).

## **5 Выводы**

Я получила навыки правильной работы с репозиториями git