

Отчёт по лабораторной работе №4

Дисциплина: Архитектура компьютеров и операционные системы

Вакутайпа Милдред

Содержание

1	Цель работы	4
2	Задание	5
3	Выполнение лабораторной работы	6
3.1	Установка git-flow	6
3.2	Установка Node.js	6
3.3	Настройка Node.js	7
3.4	Общепринятые коммиты	7
4	Выводы	15
	Список литературы	16

Список иллюстраций

3.1	включение copr	6
3.2	dnf install gitflow	6
3.3	Установка Nodejs	6
3.4	Установка pnpm	7
3.5	Запуск pnpm	7
3.6	~/bashrc	7
3.7	добавление commitizen	7
3.8	добавление standard-changelog	7
3.9	Создание git-extended	8
3.10	клонирование git-extended	8
3.11	Добавление readme.md	8
3.12	Первый коммит	8
3.13	Инициализирование pnpm	9
3.14	Заполнение пакетов	9
3.15	Добавление новых файлов	9
3.16	Инициализирование git-flow	10
3.17	Проверка ветки	10
3.18	Загрузка репозиторий	10
3.19	установка внешней ветки	10
3.20	Создание релиза	11
3.21	Добавление журнала изменений	11
3.22	Замена ветки	11
3.23	Отправка на гит	12
3.24	Создание релиза	12
3.25	Создание ветки feature_branch	12
3.26	Создание релиза с версией 1.2.3	12
3.27	Обновление номера версии	13
3.28	Создание нового журнала изменений	13
3.29	Добавление журнала	13
3.30	Замена ветки	13
3.31	Отправка данных	14
3.32	Создание релиза	14

1 Цель работы

Цель данной работы является получением навыков правильной работы с репозиториями git.

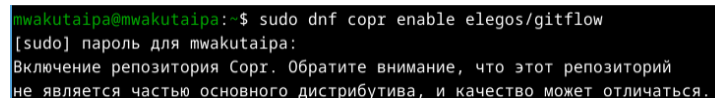
2 Задание

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

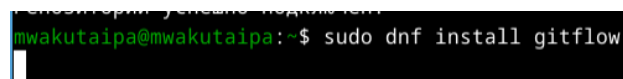
Gitflow Workflow предполагает выстраивание строгой модели ветвления с учётом выпуска проекта. Сначала я включаю репозиторий copr:



```
mwakutaipa@mwakutaipa:~$ sudo dnf copr enable elegos/gitflow
[sudo] пароль для mwakutaipa:
Включение репозитория Copr. Обратите внимание, что этот репозиторий
не является частью основного дистрибутива, и качество может отличаться.
```

Рис. 3.1: включение copr

Используя `dnf install` скачаю gitflow:



```
Репозиторий успешно подключен.
mwakutaipa@mwakutaipa:~$ sudo dnf install gitflow
```

Рис. 3.2: dnf install gitflow

3.2 Установка Node.js

Для семантического версионирования и общепринятых коммитов я устанавливаю Nodejs и npm:



```
выполнено:
mwakutaipa@mwakutaipa:~$ sudo dnf install nodejs
```

Рис. 3.3: Установка Nodejs

```
выполнено!  
mwakutaipa@mwakutaipa:~$ sudo dnf install pnpm
```

Рис. 3.4: Установка pnpm

3.3 Настройка Node.js

Запуская pnpm setup я добавляю каталог с исполняемыми файлами, устанавливаемыми yarn для работы с Node.js в переменную PATH:

```
mwakutaipa@mwakutaipa:~$ pnpm setup
```

Рис. 3.5: Запуск pnpm

Далее перелогинюсь и выполняю source ~/.bashrc:

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

Рис. 3.6: ~/.bashrc

3.4 Общепринятые коммиты

Для помощи в форматировании коммитов добавляю программу commitizen:

```
mwakutaipa@mwakutaipa:~$ pnpm add -g commitizen
```

Рис. 3.7: добавление commitizen

Добавляю standard-changelog для помощи в создании логов:

```
mwakutaipa@mwakutaipa:~$ pnpm add -g standard-changelog  
Progress: resolved 84, reused 53, downloaded 6, added 0
```

Рис. 3.8: добавление standard-changelog

Создаю репозиторий на GitHub назову его git-extended:

Create a new repository

A repository contains all project files, including the revision history. Already have a 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 *

 wakutaipa ▾

Repository name *

git-extended

✔ git-extended is available.

Great repository names are short and memorable. Need inspiration? Browse

Description (optional)

Рис. 3.9: Создание git-extended

Я клонирую его в каталог git-extended:

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

Рис. 3.10: клонирование git-extended

Создаю и добавляю файл README.md:

```
mwakutaipa@mwakutaipa:~$ cd git-extended
mwakutaipa@mwakutaipa:~/git-extended$ git add README.md
fatal: спецификатор пути «README.md» не соответствует ни одному файлу
mwakutaipa@mwakutaipa:~/git-extended$ touch README.md
mwakutaipa@mwakutaipa:~/git-extended$ git add README.md
```

Рис. 3.11: Добавление readme.md

Делаю первый коммит и выкладываю на github:

```
mwakutaipa@mwakutaipa:~/git-extended$ git commit -m "first commit"
[master (корневой коммит) 0b1a051] first commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 README.md
mwakutaipa@mwakutaipa:~/git-extended$ git remote add origin git@github.com:wakutaipa/git-extended.git
error: внешний репозиторий origin уже существует
mwakutaipa@mwakutaipa:~/git-extended$ git remote -v
origin https://github.com/wakutaipa/git-extended.git (fetch)
origin https://github.com/wakutaipa/git-extended.git (push)
mwakutaipa@mwakutaipa:~/git-extended$ git push -u origin master
```

Рис. 3.12: Первый коммит

Я инициализирую рпм:

```
mwakutaipa@mwakutaipa:~/git-extended$ pnpm init
Wrote to /home/mwakutaipa/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. 3.13: Инициализирование рпм

Заполняю несколько параметров пакета (файл package.json):

```
package.json  [-M--]  1 L: [ 1+13 14/ 15] *(341 / 342b)
{
  "name": "git-extended",
  "version": "1.0.0",
  "description": "Git repo for educational purposes",
  "main": "index.js",
  "repository": "git@github.com:wakutaipa/git-extended.git",
  "author": "wakutaipa <1832239089@pfur.ru>",
  "license": "CC-BY-4.0",
  "config": {
    "commitizen": {
      "path": "cz-conventional-changelog"
    }
  }
}
```

Рис. 3.14: Заполнение пакетов

Добавляю новые файлы и выполняю коммит (указываю тип коммит (feat)) и отправляю на гит:

```
mwakutaipa@mwakutaipa:~/git-extended$ git add .
mwakutaipa@mwakutaipa:~/git-extended$ git cz
cz-cli@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
```

Рис. 3.15: Добавление новых файлов

Инициализирую git-flow и указываю ветки:

```
mwakutaipa@mwakutaipa:~/git-extended$ git flow init
Which branch should be used for bringing forth production releases?
- master
Branch name for production releases: [master] master
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/mwakutaipa/git-extended/.git/hooks]
```

Рис. 3.16: Инициализирование git-flow

Проверяю что я на ветке develop с помощью git branch:

```
mwakutaipa@mwakutaipa:~/git-extended$ git branch
* develop
master
```

Рис. 3.17: Проверка ветки

Загружаю весь репозиторий в хранилище с помощью git push --all:

```
mwakutaipa@mwakutaipa:~/git-extended$ git push --all
```

Рис. 3.18: Загрузка репозиторий

Устанавливаю внешнюю ветку как вышестоящую для этой ветки (develop):

```
mwakutaipa@mwakutaipa:~/git-extended$ git branch --set-upstream-to=origin/develop develop
branch 'develop' set up to track 'origin/develop'.
```

Рис. 3.19: установка внешней ветки

Создаю релиз с версией 1.0.0 и журнал изменений (standard-changelog)::

```

mwakutaipa@mwakutaipa:~/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'

mwakutaipa@mwakutaipa:~/git-extended$ standard-changelog --first-release
✓ created CHANGELOG.md
✓ output changes to CHANGELOG.md
mwakutaipa@mwakutaipa:~/git-extended$

```

Рис. 3.20: Создание релиза

Добавляю журнал изменений в индекс:

```

mwakutaipa@mwakutaipa:~/git-extended$ git add CHANGELOG.md
mwakutaipa@mwakutaipa:~/git-extended$ git commit -am 'chore(site): add changelog'
[release/1.0.0 f4f1a03] chore(site): add changelog
1 file changed, 9 insertions(+)
create mode 100644 CHANGELOG.md

```

Рис. 3.21: Добавление журнала изменений

Залью релизную ветку в основную ветку:

```

mwakutaipa@mwakutaipa:~/git-extended$ git flow release finish 1.0.0
Переключились на ветку «master»
Эта ветка соответствует «origin/master».
Merge made by the 'ort' strategy.
CHANGELOG.md | 9 ++++++++
1 file changed, 9 insertions(+)
create mode 100644 CHANGELOG.md
Уже на «master»
Ваша ветка опережает «origin/master» 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 f4f1a03).

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

mwakutaipa@mwakutaipa:~/git-extended$

```

Рис. 3.22: Замена ветки

Отправляю данные на github:

```

mwakutaipa@mwakutaipa:~/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), 676 байтов | 52.00 КиБ/с, готово.
Всего 5 (изменений 2), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/wakutaipa/git-extended.git
   8fe275b..1c02dec  develop -> develop
   8fe275b..e4586ae  master -> master
mwakutaipa@mwakutaipa:~/git-extended$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 153 байта | 51.00 КиБ/с, готово.
Всего 1 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
To https://github.com/wakutaipa/git-extended.git
 * [new tag]         1.0.0 -> 1.0.0
mwakutaipa@mwakutaipa:~/git-extended$

```

Рис. 3.23: Отправка на гит

Создаю релиз на github. Для этого использую утилиты работы с github gh (gh release create):

```

mwakutaipa@mwakutaipa:~/git-extended$ gh release create v1.0.0 -F CHANGELOG.md
https://github.com/wakutaipa/git-extended/releases/tag/v1.0.0
mwakutaipa@mwakutaipa:~/git-extended$

```

Рис. 3.24: Создание релиза

Создаю ветку для новой функциональности:

```

mwakutaipa@mwakutaipa:~/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

```

Рис. 3.25: Создание ветки feature_branch

Далее, продолжаю работу с git как обычно. Создаю релиз с версией 1.2.3:

```

mwakutaipa@mwakutaipa:~/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'

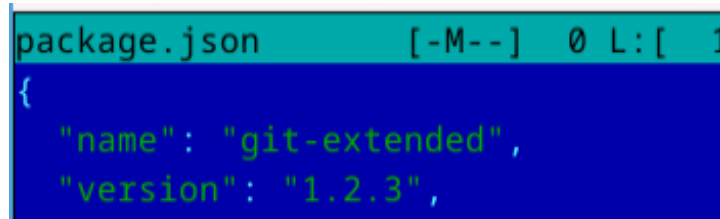
Follow-up actions:
- Bump the version number now!
- Start committing last-minute fixes in preparing your release
- When done, run:

git flow release finish '1.2.3'

```

Рис. 3.26: Создание релиза с версией 1.2.3

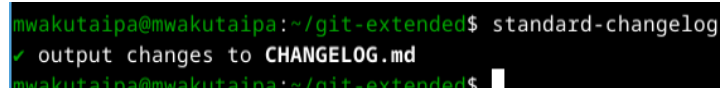
Обновляю номер версии в файле package.json в 1.2.3:



```
package.json  [-M--]  0 L: [ 1
{
  "name": "git-extended",
  "version": "1.2.3",
```

Рис. 3.27: Обновление номера версии

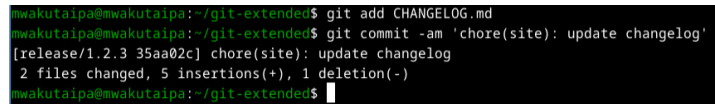
Создаю журнал изменений (standard-changelog):



```
mwakutaipa@mwakutaipa:~/git-extended$ standard-changelog
✓ output changes to CHANGELOG.md
mwakutaipa@mwakutaipa:~/git-extended$
```

Рис. 3.28: Создание нового журнала изменений

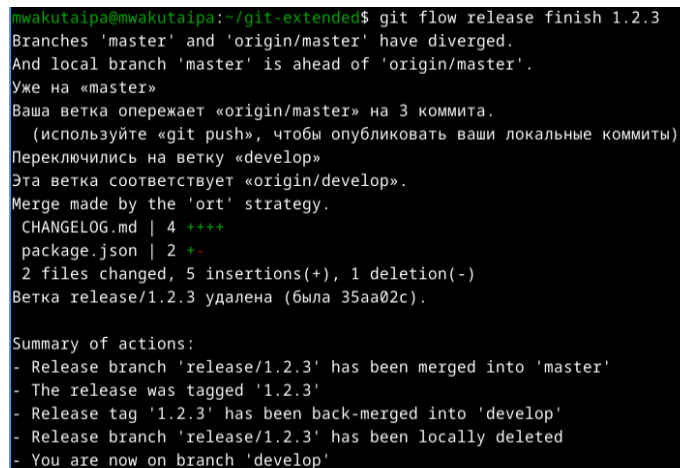
Добавляю журнал изменений в индекс:



```
mwakutaipa@mwakutaipa:~/git-extended$ git add CHANGELOG.md
mwakutaipa@mwakutaipa:~/git-extended$ git commit -am 'chore(site): update changelog'
[release/1.2.3 35aa02c] chore(site): update changelog
2 files changed, 5 insertions(+), 1 deletion(-)
mwakutaipa@mwakutaipa:~/git-extended$
```

Рис. 3.29: Добавление журнала

Залью релизную ветку в основную ветку:



```
mwakutaipa@mwakutaipa:~/git-extended$ git flow release finish 1.2.3
Branches 'master' and 'origin/master' have diverged.
And local branch 'master' is ahead of 'origin/master'.
Уже на «master»
Ваша ветка опережает «origin/master» 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 35aa02c).

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

Рис. 3.30: Замена ветки

Отправляю данные на github:

```
wakutaipa@wakutaipa:~/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), 729 байтов | 104.00 КиБ/с, готово.
Всего 6 (изменений 3), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (3/3), completed with 2 local objects.
To https://github.com/wakutaipa/git-extended.git
 1c02dec..0d9d1ca develop -> develop
 e4586ae..2b31732 master -> master
* [new branch]      feature/feature_branch -> feature/feature_branch
wakutaipa@wakutaipa:~/git-extended$ git push --tags
```

Рис. 3.31: Отправка данных

Создаю релиз на github с комментарием из журнала изменений:

```
wakutaipa@wakutaipa:~/git-extended$ gh release create v1.2.3 -F CHANGELOG.md
https://github.com/wakutaipa/git-extended/releases/tag/v1.2.3
wakutaipa@wakutaipa:~/git-extended$
```

Рис. 3.32: Создание релиза

4 Выводы

При выполнении работы я получила навыки правильной работы с репозиториями git.

Список литературы

Архитектура ЭВМ