

Отчёт по лабораторной работе 4

Архитектура Компьютера и Операционные системы

Гомес Лопес Теофания

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	8
4	Выводы	15
	Список литературы	16

Список иллюстраций

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

List of Tables

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:

```
teofaniagomeslopes@teofanialopes:~$ sudo dnf copr enable elegos/gitflow
[sudo] senha para teofaniagomeslopes:
```

Рис. 3.1: включение copr

Используя `dnf install` скачаю gitflow:

```
repositorio ativado com êxito.
teofaniagomeslopes@teofanialopes:~$ sudo dnf install gitflow
Copr repo for gitflow owned by elegos                    510  B/s | 916
Dependências resolvidas.
```

Рис. 3.2: dnf install gitflow

3.2 Установка Node.js

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

```
teofaniagomeslopes@teofanialopes:~$ sudo dnf install nodejs
Última verificação de metadados: 0:01:07 atrás em ter 04 mar 2025 12:14:48.
Dependências resolvidas.
```

Рис. 3.3: Установка Node.js

```
teofaniagomeslopes@teofanialopes:~$ sudo dnf install pnpm
Última verificação de metadados: 0:05:55 atrás em ter 04 mar 2025 12:14:48.
Dependências resolvidas.
```

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:

```
teofaniagomeslopes@teofanialopes:~$ pnpm setup
Appended new lines to /home/teofaniagomeslopes/.bashrc
```

Рис. 3.5: Запуск pnpm

Далее перелогинюсь и выполняю source ~/.bashrc:

```
to start using pnpm, run:
source /home/teofaniagomeslopes/.bashrc
teofaniagomeslopes@teofanialopes:~$ source /home/teofaniagomeslopes/.bashrc
```

Рис. 3.6: ~/.bashrc

3.4 Общепринятые коммиты

Для помощи в форматировании коммитов добавляю программу commitizen:

```
teofaniagomeslopes@teofanialopes:~$ source /home/teofaniagomeslopes/.bashrc
teofaniagomeslopes@teofanialopes:~$ pnpm add -g commitizen
```

Рис. 3.7: Добавление commitizen

Добавляю standard-changelog для помощи в создании логов:

```
Done in 23.65s
teofaniagomeslopes@teofanialopes:~$ pnpm add -g standard-changelog
WARN 2 deprecated subdependencies found: glob@7.2.3, inflight@1.0.6
Packages: +39
```

Рис. 3.8: добавление standard-changelog

Создаю репозиторий на GitHub назову его git-extended:

Owner * ltgomes1969 / Repository name * git-extended
 ✔ git-extended is available.

Great repository names are short and memorable. Need inspiration? How about...

Description (optional)

☐ Public
Anyone on the internet can see this repository. You choose who can commit.

☐ Private
You choose who can see and commit to this repository.

Рис. 3.9: Создание git-extended

Я клонирую его в каталог git-extended:

```
teofaniagomeslopes@teofanialopes: $ git clone --recursive https://github.com/ltgomes1969/git-extended.git
Cloning into 'git-extended'...
warning: You appear to have cloned an empty repository.
teofaniagomeslopes@teofanialopes: $
```

Рис. 3.10: клонирование git-extended

Создаю и добавляю файл README.md:

```
teofaniagomeslopes@teofanialopes: $ cd git-extended
teofaniagomeslopes@teofanialopes:~/git-extended$ git add README.md
fatal: pathspec 'README.md' did not match any files
teofaniagomeslopes@teofanialopes:~/git-extended$ touch README.md
teofaniagomeslopes@teofanialopes:~/git-extended$ git add README.md
teofaniagomeslopes@teofanialopes:~/git-extended$
```

Рис. 3.11: Добавление readme.md

Делаю первый коммит и выкладываю на github:

```
teofaniagomeslopes@teofanialopes:~/git-extended$ git commit -m "first commit"
[main (root-commit) 5dacce4] first commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 README.md
teofaniagomeslopes@teofanialopes:~/git-extended$ git remote add origin git@github.com:ltgomes1969/git-extended.git
error: remote origin already exists.
teofaniagomeslopes@teofanialopes:~/git-extended$ git remote -v
origin https://github.com/ltgomes1969/git-extended.git (fetch)
origin https://github.com/ltgomes1969/git-extended.git (push)
teofaniagomeslopes@teofanialopes:~/git-extended$ git push -u origin master
```

Рис. 3.12: Первый коммит

Я инициализирую rprmt:

```

branch master, set up to track origin/master.
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ npm init
Wrote to /home/teofaniagomeslopes/git-extended/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"
}
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$

```

Рис. 3.13: Инициализирование npm

Заполняю несколько параметров пакета (файл package.json):

```

package.json  [-M--] 43 L: [ 1+10 11/ 13] *(346 / 349b) 0010 0x00A
{
  "name": "git-extended",
  "version": "1.0.0",
  "description": "Git repo for educational purposes",
  "main": "index.js",
  "repository": "git@github.com:ltgomes1969/git-extended.git",
  "author": "ltgomes1969 <lopesteofania3@gmail.com>",
  "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:

```

teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ git add .
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ git cz
cz-cli@4.3.1, 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
(Move up and down to reveal more choices)

```

Рис. 3.15: Добавление новых файлов

Инициализирую git-flow и указываю ветки:

```
teofaniagomeslopes@teofanialopes:~/git-extended/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/teofaniagomeslopes/git-extended/git-extended/.git/hooks]
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$
```

Рис. 3.16: Инициализирование git-flow

Проверяю что я на ветке develop с помощью git branch:

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

Рис. 3.17: Проверка ветки

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

```
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ git push --all
Username for 'https://github.com': lfgomes1969
Password for 'https://lfgomes1969@github.com':
```

Рис. 3.18: Загрузка репозиторий

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

```
teofaniagomeslopes@teofanialopes:~/git-extended/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)::

```
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ 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 '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'

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

Рис. 3.20: Создание релиза

Добавляю журнал изменений в индекс:

```
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ git add CHANGELOG.md
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ git commit -am 'chore(site): add changelog'
[release/1.0.0 326bb64] chore(site): add changelog
1 file changed, 0 insertions(+)
create mode 100644 CHANGELOG.md
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$
```

Рис. 3.21: Добавление журнала изменений

Залью релизную ветку в основную ветку:

```
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ git push --all
Username for 'https://github.com': ltgomes1969
Password for 'https://ltgomes1969@github.com':
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 2.79 KiB | 2.79 MiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/ltgomes1969/git-extended.git
 3f26b20..451dd6e develop -> develop
 3f26b20..7f791da master -> master
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ git push --tags
Username for 'https://github.com': ltgomes1969
Password for 'https://ltgomes1969@github.com':
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 155 bytes | 155.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/ltgomes1969/git-extended.git
 * [new tag] 1.0.0 -> 1.0.0
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$
```

Рис. 3.22: Замена ветки

Отправляю данные на github:

```
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ git push --all
Username for 'https://github.com': ltgomes1969
Password for 'https://ltgomes1969@github.com':
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 2.79 KiB | 2.79 MiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/ltgomes1969/git-extended.git
 3f26b20..451dd6e develop -> develop
 3f26b20..7f791da master -> master
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ git push --tags
Username for 'https://github.com': ltgomes1969
Password for 'https://ltgomes1969@github.com':
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 155 bytes | 155.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/ltgomes1969/git-extended.git
 * [new tag] 1.0.0 -> 1.0.0
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$
```

Рис. 3.23: Отправка на гит

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

```
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ gh release create v1.0.0 -F CHANGELOG.md
https://github.com/ltgomes1969/git-extended/releases/tag/v1.0.0
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$
```

Рис. 3.24: Создание релиза

Создаю ветку для новой функциональности:

```
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ gh release create v1.0.0 -F CHANGELOG.md
https://github.com/ltgomes1969/git-extended/releases/tag/v1.0.0
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ git flow feature start feature_branch
Switched to a new branch '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

teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$
```

Рис. 3.25: Создание ветки feature_branch

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

```
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ 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, 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'

teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$
```

Рис. 3.26: Создание релиза с версией 1.2.3

Обновляю номер версии в файле package.json в 1.2.3:

```
package.json  [-M--] 19 L: [ 1+ 2 3/ 15] *(47 / 356b) 0034 0x022
{
  "name": "git-extended",
  "version": "1.2.3",
}
```

Рис. 3.27: Обновление номера версии

Создаю журнал изменений (standard-changelog):

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

Рис. 3.28: Создание нового журнала изменений

Добавляю журнал изменений в индекс:

```
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ git add CHANGELOG.md
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ git commit -am 'chore(site): update changelog'
[release/1.2.3 ece4f8f] chore(site): update changelog
2 files changed, 5 insertions(+), 1 deletion(-)
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$
```

Рис. 3.29: Добавление журнала

Залью релизную ветку в основную ветку:

```
teofaniagomeslopes@teofanialopes:~/git-extended/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'.
Already on 'master'
Your branch is ahead of 'origin/master' by 3 commits.
(use "git push" to publish your local commits)
Switched to branch 'develop'
Your branch is up to date with 'origin/develop'.
Merge made by the 'ort' strategy.
CHANGELOG.md | 4 ++++
package.json | 2 +-
2 files changed, 5 insertions(+), 1 deletion(-)
Deleted branch release/1.2.3 (was ece4f8f).

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'

teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$
```

Рис. 3.30: Замена ветки

Отправляю данные на github:

```
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ git push --all
Username for 'https://github.com': ltgomes1969
Password for 'https://ltgomes1969@github.com':
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 4 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 2.76 KiB | 2.76 MiB/s, done.
Total 6 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/ltgomes1969/git-extended.git
  451dd6e..3e2b3b7 develop -> develop
  7f791da..0100bbe master -> master
* [new branch] feature/feature_branch -> feature/feature_branch
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ git push --tags
```

Рис. 3.31: Отправка данных

Создаю релиз на github с комментарием из журнала изменений:

```
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$ gh release create v1.2.3 -F CHANGELOG.md
https://github.com/ltgomes1969/git-extended/releases/tag/v1.2.3
teofaniagomeslopes@teofanialopes:~/git-extended/git-extended$
```

Рис. 3.32: Создание релиза

4 Выводы

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

Список литературы