

Лабораторная работа №4

Операционные системы

Малюга Валерия

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	12

Список иллюстраций

3.1	Установка	7
3.2	Установка	7
3.3	Настройка	8
3.4	Настройка	8
3.5	Заполнение параметров пакета	9
3.6	Отправка новых файлов на гитхаб	9
3.7	Конфигурация git-flow	10
3.8	Конфигурация git-flow	11

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 Выполнение лабораторной работы

3.1 Установка git-flow

Установка git flow из коллекции репозитория Copr (рис. 3.1).

```
on nodejs system?
vvmalyuga@vvmalyuga:~$ sudo dnf install gitflow
Copr repo for gitflow owned by elegos          3.1 kB/s | 1.5 kB      00:00
Dependencies resolved.
=====
Package                                     Arch   Version      Repository      Size
-----
Package                                     Arch   Version      Repository      Size
```

Рис. 3.1: Установка

3.2 Установка Node.js

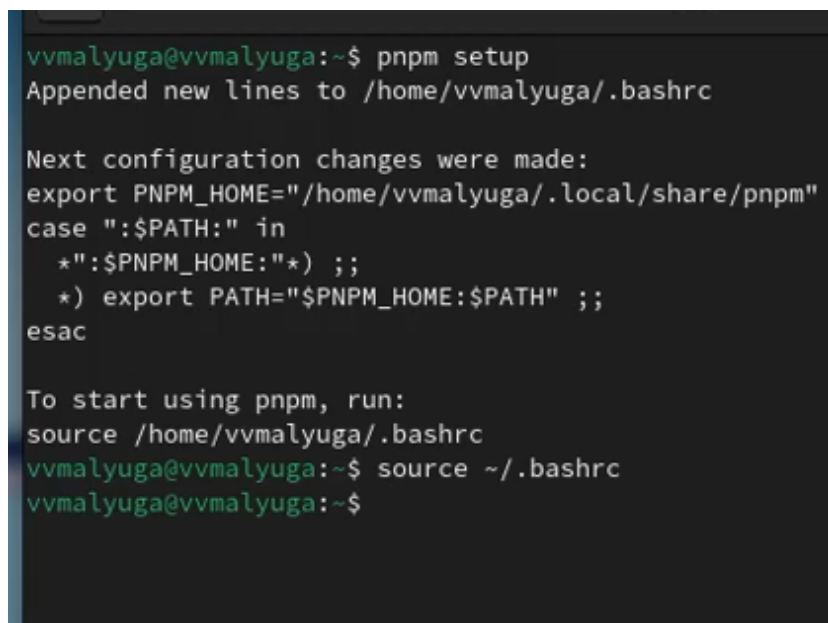
Установила nodejs и pnpm (рис. 3.2).

```
vvmalyuga@vvmalyuga:~$ sudo dnf install nodejs
Last metadata expiration check: 0:00:58 ago on Sat 09 Mar 2024 15:56:49 MSK.
Dependencies resolved.
=====
Package                                     Arch   Version      Repository      S
=====
Installing:
nodejs                                     x86_64   1:20.10.0-3.fc39   updates        4
vvmalyuga@vvmalyuga:~$ sudo yum install pnpm
Last metadata expiration check: 0:05:25 ago on Sat 09 Mar 2024 15:56:49 MSK.
Dependencies resolved.
=====
Package                                     Architecture Version      Repository
=====
Installing:
pnpm                                       noarch      8.12.0-1.fc39   updates
```

Рис. 3.2: Установка

3.3 Настройка Node.js

Для работы с Node.js добавила каталог с исполняемыми файлами, устанавливаемыми yarn, в переменную PATH. Запустила необходимую команду и перелогинилась (рис. 3.3).



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

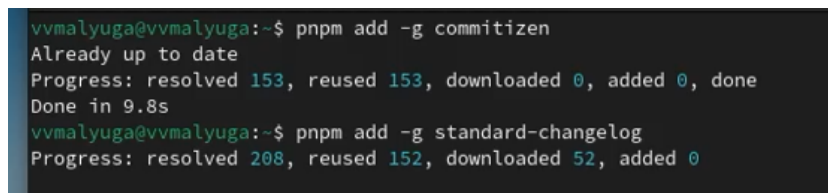
Next configuration changes were made:
export PNPM_HOME="/home/vvmalyuga/.local/share/pnpm"
case ":$PATH:" in
  *:$PNPM_HOME:* ) ;;
  *) export PATH="$PNPM_HOME:$PATH" ;;
esac

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

Рис. 3.3: Настройка

3.4 Общепринятые коммиты

Запустила программы, которые используются для помощи в форматировании коммитов и в создании логов (рис. 3.4).



```
vvmalyuga@vvmalyuga:~$ pnpm add -g commitizen
Already up to date
Progress: resolved 153, reused 153, downloaded 0, added 0, done
Done in 9.8s
vvmalyuga@vvmalyuga:~$ pnpm add -g standard-changelog
Progress: resolved 208, reused 152, downloaded 52, added 0
```

Рис. 3.4: Настройка

Создала репозиторий, сделала первый коммит и загрузила на гитхаб. Далее

заполняю параметры пакета (рис. 3.5).

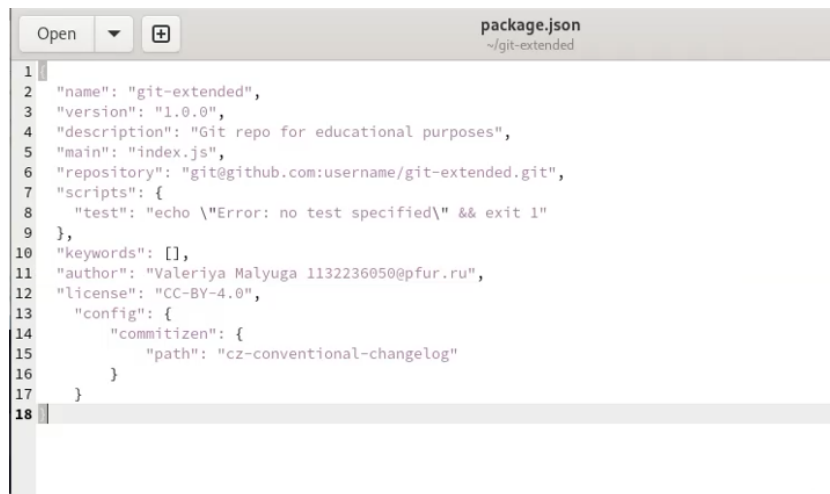


Рис. 3.5: Заполнение параметров пакета

Добавила новые файлы, выполнила коммит, отправила на гитхаб (рис. 3.6).

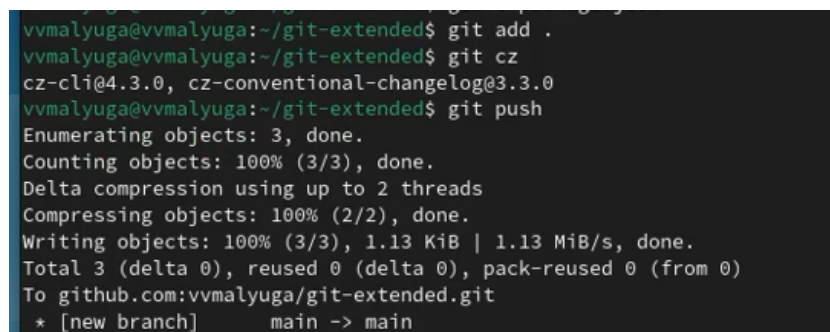


Рис. 3.6: Отправка новых файлов на гитхаб

Инициализировала git-flow. Префикс для ярлыков установила v. Проверила, что я на ветке develop. Загрузила весь репозиторий в хранилище. Установила внешнюю ветку как вышестоящую для этой ветки. Создала релиз с версией 1.0.0. Создала журнал изменений (рис. 3.7).

```

vvmalyuga@vvmalyuga:~/git-extended$ git flow init

Which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main]
main
vvmalyuga@vvmalyuga:~/git-extended$ git push --all
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:   https://github.com/vvmalyuga/git-extended/pull/new/develop
remote:
To github.com:vvmalyuga/git-extended.git
* [new branch]      develop -> develop
vvmalyuga@vvmalyuga:~/git-extended$ git branch --set-upstream-to=origin/develop develop
branch 'develop' set up to track 'origin/develop'.
vvmalyuga@vvmalyuga:~/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'

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

```

Рис. 3.7: Конфигурация git-flow

Создала ветку для новой функциональности. По окончании разработки новой функциональности следующим шагом следует объединить ветку. Создала релиз с версией 1.2.3. Обновила номер версии в файле package.json. Установила её в 1.2.3. Создала журнал изменений. Добавим журнал изменений в индекс. Залила релизную ветку в основную ветку. Отправила данные на github. Создала релиз на github с комментарием из журнала изменений (рис. 3.8).

```

vvmalyuga@vvmalyuga:~/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

- Start committing last-minute fixes in preparing your release
- When done, run:

    git flow release finish '1.0.0'
Fatal: There is an existing release branch '1.0.0'. Finish that one first.
vvmalyuga@vvmalyuga:~/git-extended$ git flow release finish 1.0.0
Branches 'develop' and 'origin/develop' have diverged.
And local branch 'develop' is ahead of 'origin/develop'.
Already on 'main'
Your branch is up to date with 'origin/main'.
Switched to branch 'develop'
Your branch is ahead of 'origin/develop' by 1 commit.
    (use "git push" to publish your local commits)
Auto-merging CHANGELOG.md
CONFLICT (add/add): Merge conflict in CHANGELOG.md
Automatic merge failed; fix conflicts and then commit the result.
Fatal: There were merge conflicts.
vvmalyuga@vvmalyuga:~/git-extended$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 2 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 1.07 KiB | 1.07 MiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack reused 0 (from 0)
To github.com:vvmalyuga/git-extended.git
   5097ff6..2165ab6  develop -> develop
vvmalyuga@vvmalyuga:~/git-extended$ gedit package.json
vvmalyuga@vvmalyuga:~/git-extended$ git flow release start 1.2.3
Fatal: There is an existing release branch '1.0.0'. Finish that one first.
vvmalyuga@vvmalyuga:~/git-extended$ git add CHANGELOG.md
vvmalyuga@vvmalyuga:~/git-extended$ git commit -am 'chore(site): update changelog'

```

Рис. 3.8: Конфигурация git-flow

4 Выводы

Получила навыки правильной работы с репозиториями git.