

Отчёт по лабораторной работе №4

Дисциплина: Операционные системы

Филатов Илья Гурамович

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	Настройки Node.js	7
3.4	Практический сценарий использования git	8
3.4.1	Создание репозитория github	8
3.5	Конфигурация git-flow	9
3.6	Работа с репозиторием git-flow	10
4	Выводы	13
5	Список литературы	14

Список иллюстраций

3.1	Установка пакета gitflow	6
3.2	Установка пакетов 2	7
3.3	Настройка Nodejs	7
3.4	Установка доп. пакетов	8
3.5	Инициализация репозитория	8
3.6	Конфигурация по стандарту cz	9
3.7	Инициализация git flow	9
3.8	Настройка ветки develop	10
3.9	Создание релиза	10
3.10	Создание ветки новой функциональности	11
3.11	Изменение версии	11
3.12	Создание тестового релиза 1.2.3	11

1 Цель работы

- Получить навыки правильной работы с репозиториями git

2 Задание

1. Выполнить работу для тестового репозитория
 - Преобразовать рабочий репозиторий в репозиторий с git-flow и conventional commits

3 Выполнение лабораторной работы

3.1 Установка git-flow

Установим gitflow из коллекции репозитория Copr (рис. 3.1).

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

Проект Fedora не имеет какого-либо влияния на содержимое этого
репозитория за рамками правил, описанных в Вопросах и Ответах Copr в
<https://docs.pagure.org/copr.copr/user\_documentation.html#what-i-can-build-in-copr>,
а качество и безопасность пакетов не поддерживаются на каком-либо уровне.

Не отправляйте сообщения об ошибках этих пакетов в Fedora
Bugzilla. В случае возникновения проблем обращайтесь к владельцу этого репозитория.

Do you really want to enable copr.fedorainfracloud.org/elegos/gitflow? [y/N]: y
Репозиторий успешно подключен.
[igfilatov@igfilatov tmp]$ sudo dnf install gitflow
```

Рис. 3.1: Установка пакета gitflow

3.2 Установка Node.js

Также установим Node.js на котором базируется ПО для семантического версионирования коммитов (рис. 3.2).

```
[igfilatov@igfilatov tmp]$ sudo dnf install pnpm
Последняя проверка окончания срока действия метаданных: 0:02:05 назад, Чт 06 мар 2025 2
Зависимости разрешены.
=====
Пакет          Архитектура      Версия           Репозиторий
=====
Установка:
pnpm           noarch           9.0.6-1.fc40     updates
=====
Результат транзакции
=====
Установка 1 Пакет

Объем загрузки: 2.8 М
Объем изменений: 14 М
Продолжить? [д/н]: y
Загрузка пакетов...
```

Рис. 3.2: Установка пакетов 2

3.3 Настройки Node.js

Используя команду `pnpm setup` и `source ~/.bashrc` добавляем каталог с исполняемыми файлами от yarn в переменную `PATH` (рис. 3.3).

```
[igfilatov@igfilatov tmp]$ pnpm setup
Appended new lines to /home/igfilatov/.bashrc

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

To start using pnpm, run:
source /home/igfilatov/.bashrc
[igfilatov@igfilatov tmp]$ source ~/.bashrc
[igfilatov@igfilatov tmp]$
```

Рис. 3.3: Настройка Node.js

Далее установим пакет `commitizen` для помощи в форматировании коммитов и пакет `standard-changelog` который используется для помощи в создании логов (рис. 3.4).

```
Follow @pnpmjs for updates: https://twitter.com/pnpmjs

WARN 2 deprecated subdependencies found: glob@7.2.3, inflight@1.0.6
Packages: +151
Progress: resolved 151, reused 0, downloaded 151, added 151, done

/home/igfilatov/.local/share/pnpm/global/5:
+ commitizen 4.3.1

Done in 4.5s
[igfilatov@igfilatov tmp]$ pnpm add -g standard-changelog
WARN 2 deprecated subdependencies found: glob@7.2.3, inflight@1.0.6
Packages: +39
Progress: resolved 190, reused 151, downloaded 39, added 39, done

/home/igfilatov/.local/share/pnpm/global/5:
+ standard-changelog 6.0.0

Done in 3.7s
[igfilatov@igfilatov tmp]$
```

Рис. 3.4: Установка доп. пакетов

3.4 Практический сценарий использования git

3.4.1 Создание репозитория github

Создадим учебный репозиторий на github и сделаем первый коммит (рис. 3.5).

```
[igfilatov@igfilatov tmp]$ git init
Инициализирован пустой репозиторий Git в /home/igfilatov/Documents/tmp/.git/
[igfilatov@igfilatov tmp]$ git add .
[igfilatov@igfilatov tmp]$ git commit -m "first commit"
[master (корневой коммит) 4135c21] first commit
1 file changed, 1 insertion(+)
create mode 100644 README.md
[igfilatov@igfilatov tmp]$ git branch -M master
[igfilatov@igfilatov tmp]$ git remote add origin https://ghp_10oqWwpDJ09rDRUB8fSUW
github.com/igfilatov/git-extended.git
[igfilatov@igfilatov tmp]$ git push -u origin master
Перечисление объектов: 3, готово.
Подсчет объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 241 байт | 241.00 КиБ/с, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/igfilatov/git-extended.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
[igfilatov@igfilatov tmp]$
```

Рис. 3.5: Инициализация репозитория

Проведём конфигурацию общепринятых коммитов и сделаем повторный коммит (рис. 3.6).


```

? Select the type of change that you're committing: chore:   Other changes that don't mod
or test files
? What is the scope of this change (e.g. component or file name): (press enter to skip)
packages.json
? Write a short, imperative tense description of the change (max 78 chars):
(33) added commitizen to packages.json
? 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
[master eldec6b] chore(packages.json): added commitizen to packages.json
1 file changed, 14 insertions(+)
create mode 100644 package.json
[igfilatov@igfilatov tmp]$ git push
Перечисление объектов: 4, готово.
Подсчет объектов: 100% (4/4), готово.
При сжатии изменений используется до 8 потоков
Сжатие объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 588 байтов | 588.00 КиБ/с, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/igfilatov/git-extended.git
4135c21..eldec6b master -> master
[igfilatov@igfilatov tmp]$

```

Рис. 3.6: Конфигурация по стандарту cz

3.5 Конфигурация git-flow

Инициализируем git-flow устанавливая префикс для ярлыков в v, проверим что мы на ветке develop и загрузим весь репозиторий в хранилище (рис. 3.7).

```

- master
Branch name for production releases: [master]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? [] v
Hooks and filters directory? [/home/igfilatov/Documents/tmp/.git/hooks]
[igfilatov@igfilatov tmp]$ git branch
* develop
master
[igfilatov@igfilatov tmp]$ 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/igfilatov/git-extended/pull/new/develop
remote:
To https://github.com/igfilatov/git-extended.git
 * [new branch]      develop -> develop
[igfilatov@igfilatov tmp]$

```

Рис. 3.7: Инициализация git flow

Установим внешнюю ветку как вышестоящую для этой ветки и создадим релиз с версией 1.0.0 и журнал изменений (рис. 3.8).

```
[igfilatov@igfilatov tmp]$ git branch --set-upstream-to=origin/develop develop
branch 'develop' set up to track 'origin/develop'.
[igfilatov@igfilatov tmp]$ 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'

[igfilatov@igfilatov tmp]$ standard-changelog --first-release
✓ created CHANGELOG.md
✓ output changes to CHANGELOG.md
[igfilatov@igfilatov tmp]$
```

Рис. 3.8: Настройка ветки develop

Добавим изменения в индекс, зельём релизную ветвь в основную, отправим данные и создадим релиз на github (рис. 3.9).

```
[igfilatov@igfilatov tmp]$ git push --all
Перечисление объектов: 6, готово.
Подсчет объектов: 100% (6/6), готово.
При сжатии изменений используется до 8 потоков
Сжатие объектов: 100% (4/4), готово.
Запись объектов: 100% (5/5), 592 байта | 592.00 КиБ/с, готово.
Total 5 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/igfilatov/git-extended.git
   e1dec6b..9628e42  develop -> develop
   e1dec6b..e4e73c4  master -> master
[igfilatov@igfilatov tmp]$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 188 байтов | 188.00 КиБ/с, готово.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/igfilatov/git-extended.git
 * [new tag]         v1.0.0 -> v1.0.0
[igfilatov@igfilatov tmp]$ gh release create v1.0.0 -F CHANGELOG.md
To get started with GitHub CLI, please run:  gh auth login
Alternatively, populate the GH_TOKEN environment variable with a GitHub API authentication
[igfilatov@igfilatov tmp]$ gh release create v1.0.0 -F CHANGELOG.md
https://github.com/igfilatov/git-extended/releases/tag/v1.0.0
[igfilatov@igfilatov tmp]$
```

Рис. 3.9: Создание релиза

3.6 Работа с репозиторием git-flow

Для демонстрации создадим ветку новой функциональности и добавим релиз с версией 1.2.3 (рис. 3.10).

```

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

[igfilatov@igfilatov tmp]$ 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'

[igfilatov@igfilatov tmp]$

```

Рис. 3.10: Создание ветки новой функциональности

Обновим номер версии в файле `package.json`, создадим журнал изменений и добавим в индекс (рис. 3.11).

```

[igfilatov@igfilatov tmp]$ nano package.json
[igfilatov@igfilatov tmp]$ standard-changelog
✓ output changes to CHANGELOG.md
[igfilatov@igfilatov tmp]$ git add CHANGELOG.md

```

Рис. 3.11: Изменение версии

Сделаем коммит, снова заальём ветку в основную и после отправки данных на github создаём релиз (рис. 3.12).

```

[igfilatov@igfilatov tmp]$ git push --all
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
При сжатии изменений используется до 8 потоков
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 834 байта | 834.00 КиБ/с, готово.
Total 6 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To https://github.com/igfilatov/git-extended.git
   9628e42..44b64d5  develop -> develop
   e4e73c4..2607b64  master -> master
[igfilatov@igfilatov tmp]$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 190 байтов | 190.00 КиБ/с, готово.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/igfilatov/git-extended.git
 * [new tag]         v1.2.3 -> v1.2.3
[igfilatov@igfilatov tmp]$ gh release create v1.2.3 -F CHANGELOG.md
https://github.com/igfilatov/git-extended/releases/tag/v1.2.3
[igfilatov@igfilatov tmp]$

```

Рис. 3.12: Создание тестового релиза 1.2.3

В конце концов преобразуем наш рабочий репозиторий в репозиторий с git-flow и conventional commits

4 Выводы

Я приобрёл навыки правильной работы с репозиториями git, git-flow и conventional commits

5 Список литературы

1. Архитектура ЭВМ