

Отчёт по лабораторной работе №4

Дисциплина: операционные системы

Казаазев Даниил Михайлович

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	16

Список иллюстраций

3.1	Установка git flow	7
3.2	Установка Node.js и npm	7
3.3	Настройка	8
3.4	Установка программы для форматирования коммитов	8
3.5	Установка программы для создания логов	8
3.6	Создание репозитория	8
3.7	Создание директории для репозитория	9
3.8	Создание директории для репозитория	9
3.9	Создание директории для репозитория	9
3.10	Созданный репозиторий	10
3.11	Конфигурация файла package.json до его настройки	10
3.12	Конфигурация файла package.json после его настройки	10
3.13	Добавление изменений	10
3.14	Выполнение коммита	11
3.15	Отправка файлов на GitHub	11
3.16	Инициализация git flow и проверка ветки develop	11
3.17	Загрузка репозитория в хранилище	12
3.18	Создание релиза	12
3.19	Создание журнала изменений	12
3.20	Добавление журнала изменений в индекс	12
3.21	Заливка релизной ветки	13
3.22	Отправка на GitHub	13
3.23	Создание релиза	13
3.24	Создание ветки для функциональности	14
3.25	Окончание ветки функциональности	14
3.26	Заливка релизной ветки в основную	14
3.27	Отправка на GitHub	15

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

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

```
[dmkazazaev@dmkazazaev ~]$ sudo dnf copr enable elegos/gitflow
[sudo] пароль для dmkazazaev:
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.
[dmkazazaev@dmkazazaev ~]$ sudo dnf install gitflow
Copr repo for gitflow owned by elegos          11 kB/s | 3.3 kB      00:00
Пакет gitflow-1.12.3-1.fc34.x86_64 уже установлен.
Зависимости разрешены.
Нет действий для выполнения.
Выполнено!
```

Рис. 3.1: Установка git flow

Устанавливаю Node.js и npm. (рис. 3.2)

```
[dmkazazaev@dmkazazaev ~]$ sudo dnf install nodejs
Последняя проверка окончания срока действия метаданных: 0:00:12 назад, Сб 09 мар
 2024 18:54:38.
Пакет nodejs-1:18.19.0-1.fc38.x86_64 уже установлен.
Зависимости разрешены.
Нет действий для выполнения.
Выполнено!
[dmkazazaev@dmkazazaev ~]$ sudo npm install -g npm

changed 1 package in 1s

1 package is looking for funding
  run `npm fund` for details
```

Рис. 3.2: Установка Node.js и npm

Настраиваю Node.js. (рис. 3.3)

```
[dmkazazaev@dmkazazaev ~]$ pnpm setup
No changes to the environment were made. Everything is already up to date.
[dmkazazaev@dmkazazaev ~]$ source ~/.bashrc
```

Рис. 3.3: Натсройка

Устанавливаю программу для форматирования коммитов. (рис. 3.4)

```
[root@dmkazazaev ~]# sudo npm install -g cz-git commitizen
added 169 packages in 10s
27 packages are looking for funding
  run `npm fund` for details
```

Рис. 3.4: Установка программы для форматирования коммитов

Устанавливаю программу для создания логов. (рис. 3.5)

```
[root@dmkazazaev ~]# npm install -g standard-changelog
added 71 packages in 8s
16 packages are looking for funding
  run `npm fund` for details
```

Рис. 3.5: Установка программы для создания логов

Создаю новый репозиторий на GitHub. (рис. 3.6)

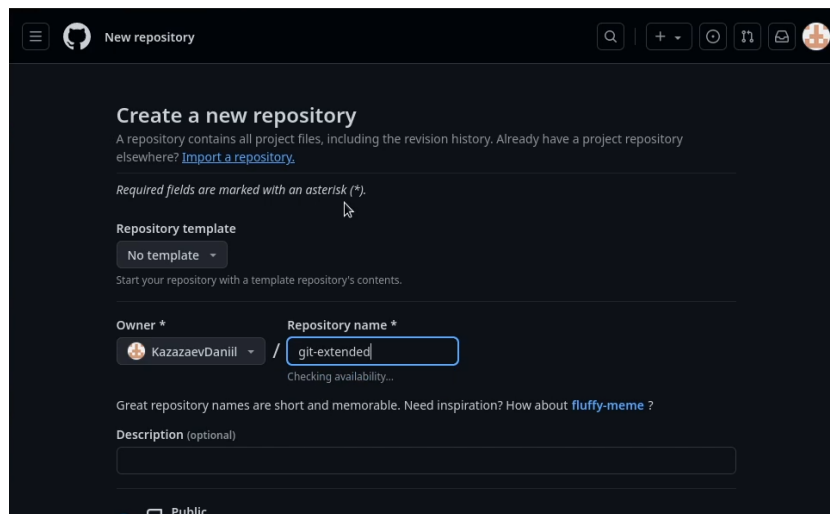


Рис. 3.6: Создание репозитория

Создаю директорию для репозитория.(рис. 3.7)


```
[root@dmkazazaev ~]# cd work/study/2023-2024/Операционные системы/
[root@dmkazazaev Операционные системы]# mkdir git-extended
[root@dmkazazaev Операционные системы]# cd git-extended/
```

Рис. 3.7: Создание директории для репозитория

Создаю репозиторий по инструкции.(тут 2 скрина)(рис. 3.8)

```
[root@dmkazazaev git-extended]# echo "# git-extended" >> README.md
[root@dmkazazaev git-extended]# git init
Инициализирован пустой репозиторий Git в /root/work/study/2023-2024/Операционные
системы/git-extended/.git/
[root@dmkazazaev git-extended]# git commit -m "first commit"
Текущая ветка: master

Начальный коммит

Неотслеживаемые файлы:
(используйте «git add <файл>...», чтобы добавить в то, что будет включено в ко
ммит)

README.md

индекс пуст, но есть неотслеживаемые файлы
(используйте «git add», чтобы проиндексировать их)
[root@dmkazazaev git-extended]# git add .
[root@dmkazazaev git-extended]# git commit -m "first commit"
[master (корневой коммит) d9216b2] first commit
1 file changed, 1 insertion(+)
create mode 100644 README.md
```

Рис. 3.8: Создание директории для репозитория

```
[root@dmkazazaev git-extended]# git remote add origin git@github.com:KazazaevDan
iil/git-extended.git
[root@dmkazazaev git-extended]# git push -u origin master
Перечисление объектов: 3, готово.
Подсчет объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 234 байта | 234.00 КиБ/с, готово.
Всего 3 (изменений 0), повторно использовано 0 (изменений 0), повторно использов
ано пакетов 0
To github.com:KazazaevDaniil/git-extended.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

Рис. 3.9: Создание директории для репозитория

Созданный репозиторий.(рис. 3.10)

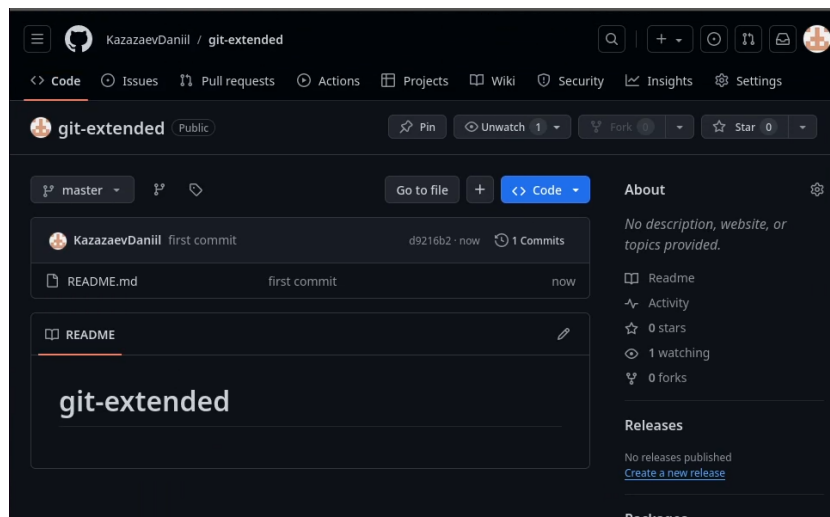


Рис. 3.10: Созданный репозиторий

Конфигурация файла package.json до его настройки.(рис. 3.11)

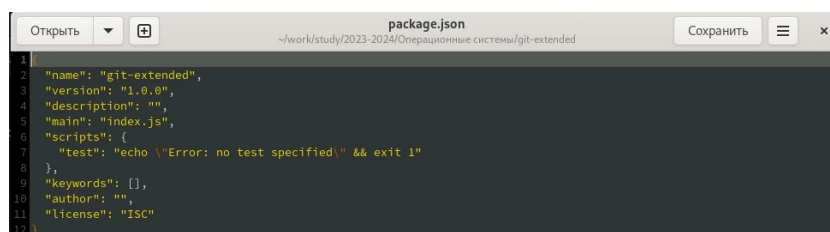


Рис. 3.11: Конфигурация файла package.json до его настройки

Конфигурация файла package.json после настройки.(рис. 3.12)

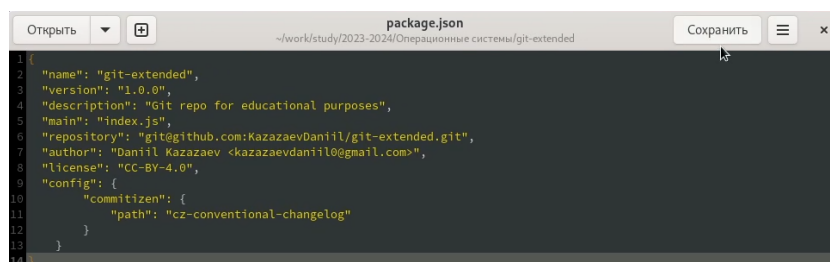


Рис. 3.12: Конфигурация файла package.json после его настройки

Добавляю изменения.(рис. 3.13)

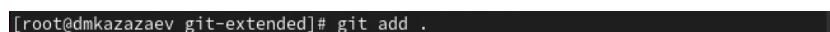


Рис. 3.13: Добавление изменений

Выполняю коммит командой `gh cz.`(рис. 3.14)

```
[root@dmkazazaev 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
(Move up and down to reveal more choices)
```

Рис. 3.14: Выполнение коммита

Отправляю файлы на GitHub.(рис. 3.15)

```
[root@dmkazazaev git-extended]# git push
Перечисление объектов: 4, готово.
Подсчет объектов: 100% (4/4), готово.
При сжатии изменений используется до 3 потоков
Сжатие объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 514 байтов | 514.00 КиБ/с, готово.
Всего 3 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
To github.com:KazazaevDaniil/git-extended.git
d9216b2..004fab9 master -> master
```

Рис. 3.15: Отправка файлов на GitHub

Инициализирую git flow и проверяю ветку develop.(рис. 3.16)

```
[root@dmkazazaev 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? [] v
Hooks and filters directory? [/root/work/study/2023-2024/Операционные системы/git-extended/.git/hooks]
[root@dmkazazaev git-extended]# git branch
* develop
master
```

Рис. 3.16: Инициализация git flow и проверка ветки develop

Загружаю весь репозиторий в хранилище и устанавливаю внешнюю ветку как вышестоящую.(рис. 3.17).

```
[root@dmkazazaev git-extended]# git push --all
Всего 0 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:   https://github.com/KazazaevDaniil/git-extended/pull/new/develop
remote:
To github.com:KazazaevDaniil/git-extended.git
 * [new branch]      develop -> develop
[root@dmkazazaev git-extended]# git branch --set-upstream-to=origin/develop develop
branch 'develop' set up to track 'origin/develop'.
```

Рис. 3.17: Загрузка репозитория в хранилище

Создаю релиз 1.0.0.(рис. 3.18)

```
[root@dmkazazaev 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'
```

Рис. 3.18: Создание релиза

Создаю журнал изменений.(рис. 3.19)

```
[root@dmkazazaev git-extended]# standard-changelog --first-release
✓ created CHANGELOG.md
✓ output changes to CHANGELOG.md
```

Рис. 3.19: Создание журнала изменений

Добавляю журнал изменений в индекс.(рис. 3.20)

```
[root@dmkazazaev git-extended]# git add CHANGELOG.md
[root@dmkazazaev git-extended]# git commit -am 'chore(site): add changelog'
```

Рис. 3.20: Добавление журнала изменений в индекс

Заливаю основную ветку.(рис. 3.21)

```
[root@dmkazazaev git-extended]# git flow release finish 1.0.0
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 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 d39bb14).

Summary of actions:
- Release branch 'release/1.0.0' has been merged into 'master'
- The release was tagged 'v1.0.0'
- Release tag 'v1.0.0' has been back-merged into 'develop'
- Release branch 'release/1.0.0' has been locally deleted
- You are now on branch 'develop'
```

Рис. 3.21: Заливка релизной ветки

Отправляю данные на GitHub.(рис. 3.22)

```
[root@dmkazazaev git-extended]# git push --all
Перечисление объектов: 6, готово.
Подсчет объектов: 100% (6/6), готово.
При сжатии изменений используется до 3 потоков
Сжатие объектов: 100% (5/5), готово.
Запись объектов: 100% (5/5), 672 байта | 672.00 КиБ/с, готово.
Всего 5 (изменений 2), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (2/2), done.
To github.com:KazazaevDaniil/git-extended.git
 004fab9..9fd8a1b develop -> develop
 004fab9..99c9b27 master -> master
[root@dmkazazaev git-extended]# git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 180 байтов | 180.00 КиБ/с, готово.
Всего 1 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
To github.com:KazazaevDaniil/git-extended.git
 * [new tag]          v1.0.0 -> v1.0.0
```

Рис. 3.22: Отправка на GitHub

Создаю релиз на GitHub.(рис. 3.23)

```
[root@dmkazazaev git-extended]# gh release create v1.0.0 -F CHANGELOG.md
https://github.com/KazazaevDaniil/git-extended/releases/tag/v1.0.0
```

Рис. 3.23: Создание релиза

Создаю новую ветку для функциональности.(рис. 3.24)

```
[root@dmkazazaev 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. 3.24: Создание ветки для функциональности

Окончание ветки для функциональности.(рис. 3.25)

```
[root@dmkazazaev git-extended]# git flow feature finish feature_branch
Переключились на ветку «develop»
Эта ветка соответствует «origin/develop».
Уже актуально.
Ветка feature/feature_branch удалена (была 9fd8a1b).

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

Рис. 3.25: Окончание ветки функциональности

Заливаю ветку в основную ветку.(рис. 3.26)

```
[root@dmkazazaev 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: Заливка релизной ветки в основную

Отправляю все на GitHub.(рис. 3.27)

```
[root@dmkazazaev git-extended]# git push --all
Перечисление объектов: 2, готово.
Подсчет объектов: 100% (2/2), готово.
При сжатии изменений используется до 3 потоков
Сжатие объектов: 100% (2/2), готово.
Запись объектов: 100% (2/2), 369 байтов | 369.00 КиБ/с, готово.
Всего 2 (изменений 1), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (1/1), done.
To github.com:KazazaevDaniil/git-extended.git
   9fd8a1b..bffd22a  develop -> develop
   99c9b27..838a2e2  master -> master
[root@dmkazazaev git-extended]# git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 185 байтов | 185.00 КиБ/с, готово.
Всего 1 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
To github.com:KazazaevDaniil/git-extended.git
 * [new tag]         v1.2.3 -> v1.2.3
```

Рис. 3.27: Отправка на GitHub

4 Выводы

В результате выполнения лабораторной работы я получил навыки правильной работы с репозиториями git.