

# **Отчёт по лабораторной работе №4**

**Операционные системы**

Мурашов Иван Вячеславович

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>7</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>8</b>
4.1	Установка программного обеспечения . . . . .	8
4.2	Общепринятые коммиты . . . . .	9
<b>5</b>	<b>Выводы</b>	<b>15</b>

## Список иллюстраций

4.1	Установка git-flow . . . . .	8
4.2	Установка Node.js . . . . .	8
4.3	Настройка Node.js . . . . .	9
4.4	Установка ПО . . . . .	9
4.5	Установка ПО . . . . .	9
4.6	Клонирование репозитория . . . . .	10
4.7	Настройка репозитория . . . . .	10
4.8	Настройка конфигураций . . . . .	10
4.9	Отправка файлов на сервер . . . . .	11
4.10	Инициализация git-flow . . . . .	11
4.11	Отправка файлов на сервер . . . . .	12
4.12	Работа с ветками . . . . .	12
4.13	Работа с ветками . . . . .	12
4.14	Отправка файлов на сервер . . . . .	13
4.15	Создание релиза . . . . .	13
4.16	Создание и объединение веток . . . . .	13
4.17	Создание релиза . . . . .	14
4.18	Настройка конфигураций . . . . .	14
4.19	Работа с ветками . . . . .	14

## Список таблиц

# 1 Цель работы

Целью данной лабораторной работы является 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 Теоретическое введение

Gitflow Workflow опубликована и популяризована Винсентом Дриссеном.

Gitflow Workflow предполагает выстраивание строгой модели ветвления с учётом выпуска проекта.

Данная модель отлично подходит для организации рабочего процесса на основе релизов.

Работа по модели Gitflow включает создание отдельной ветки для исправлений ошибок в рабочей среде.

Последовательность действий при работе по модели Gitflow:

- Из ветки master создаётся ветка develop.
- Из ветки develop создаётся ветка release.
- Из ветки develop создаются ветки feature.
- Когда работа над веткой feature завершена, она сливается с веткой develop.
- Когда работа над веткой релиза release завершена, она сливается в ветки develop и master.
- Если в master обнаружена проблема, из master создаётся ветка hotfix.
- Когда работа над веткой исправления hotfix завершена, она сливается в ветки develop и master.

## 4 Выполнение лабораторной работы

### 4.1 Установка программного обеспечения

Устанавливаю git-flow (рис. 4.1).

```
ivmurashov@ivmurashov:~$ sudo dnf install gitflow
Copr repo for gitflow owned by elegos 1.3 kB/s | 1.5 kB 00:01
Зависимости разрешены.
=====
Пакет Архитектура
Версия Репозиторий Размер
=====
Установка:
gitflow
x86_64 1.12.3-1.fc34 copr:copr.fedorainfracloud.org:elegos:gitflow 57 k
Результат транзакции
=====
Установка 1 Пакет
Объем загрузки: 57 k
Объем изменений: 262 k
```

Рис. 4.1: Установка git-flow

Устанавливаю Node.js (рис. 4.2).

```
root@ivmurashov:~# dnf install npm
Последняя проверка окончания срока действия метаданных: 0:02:40 назад, Пт 08 мар
2024 18:03:29.
Зависимости разрешены.
=====
Пакет Архитектура Версия Репозиторий Размер
=====
Установка:
npm noarch 8.12.0-1.fc39 updates 2.6 M
Результат транзакции
=====
Установка 1 Пакет
```

Рис. 4.2: Установка Node.js

Настраиваю Node.js (рис. 4.3).



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

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

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

Рис. 4.3: Настройка Node.js

## 4.2 Общепринятые коммиты

Устанавливаю программу для помощи в форматировании коммитов (рис. 4.4).

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

Update available! 8.12.0 → 8.15.4.
Changelog: https://github.com/pnpm/pnpm/releases/tag/v8.15.4
Run "pnpm add -g pnpm" to update.

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

Progress: resolved 93, reused 0, downloaded 81, added 0
```

Рис. 4.4: Установка ПО

Устанавливаю программу для помощи в создании логов (рис. 4.5).

```
ivmurashov@ivmurashov:~$ pnpm add -g standard-changelog
Packages: +56
Progress: resolved 209, reused 153, downloaded 56, added 56, done

/home/ivmurashov/.local/share/pnpm/global/5:
+ standard-changelog 5.0.0

Done in 9.9s
```

Рис. 4.5: Установка ПО

Создаю и клонирую репозиторий (рис. 4.6).

```
ivmurashov@ivmurashov:~/work$ git clone --recursive git@github.com:neve7mind/git-extended.git
Клонирование в «git-extended»...
warning: Похоже, что вы клонировали пустой репозиторий.
```

Рис. 4.6: Клонирование репозитория

Настраиваю репозиторий, делаю 1ый коммит и выкладываю его на github (рис. 4.7).

```
ivmurashov@ivmurashov:~$ echo "# git-extended" >> README.md
ivmurashov@ivmurashov:~$ git init
Инициализирован пустой репозиторий Git в /home/ivmurashov/.git/
ivmurashov@ivmurashov:~$ git add README.md
ivmurashov@ivmurashov:~$ git commit -m "first commit"
[master (корневой коммит) 30b323b] first commit
1 file changed, 1 insertion(+)
create mode 100644 README.md
ivmurashov@ivmurashov:~$ git branch -M main
ivmurashov@ivmurashov:~$ git remote add origin git@github.com:neve7mind/git-extended.git
ivmurashov@ivmurashov:~$ git push -u origin main
Перечисление объектов: 3, готово.
Подсчет объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 230 байтов | 230.00 КиБ/с, готово.
Всего 3 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
To github.com:neve7mind/git-extended.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
```

Рис. 4.7: Настройка репозитория

Настраиваю конфигурацию для пакетов Node.js (рис. 4.8).

```
package.json  [-----]  9 L: [ 1+11 12/ 15] *(377 / 386b) 0010 0x00A [*][X]
{
  "name": "git-extended",
  "version": "1.0.0",
  "description": "Git repo for educational purposes",
  "main": "index.js",
  "repository": "git@github.com:neve7mind/git-extended.git",
  "author": "neve7mind ivan.murashov1200@gmail.com",
  "license": "CC-BY-4.0",
  "config": {
    "commitizen": {
      "path": "cz-conventional-changelog"
    }
  }
}
```

Рис. 4.8: Настройка конфигураций

Добавляю новые файлы, выполняю коммит и отправляю файлы на сервер (рис. 4.9).

```

ivmurashov@ivmurashov:~/work/git-extended$ git add .
ivmurashov@ivmurashov:~/work/git-extended$ git cz
cz-cli@4.3.0, cz-conventional-changelog@3.3.0

? Select the type of change that you're committing: feat: A new feature
? What is the scope of this change (e.g. component or file name): (press enter to skip) package.json
? Write a short, imperative tense description of the change (max 80 chars):
  (29) add configuration for package
? 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
[main 4bb361c] feat(package.json): add configuration for package
1 file changed, 14 insertions(+)
create mode 100644 package.json
ivmurashov@ivmurashov:~/work/git-extended$ git push
Перечисление объектов: 4, готово.
Подсчет объектов: 100% (4/4), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 531 байт | 531.00 КиБ/с, готово.
Всего 3 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0

```

Рис. 4.9: Отправка файлов на сервер

Инициализирую git-flow (рис. 4.10).

```

ivmurashov@ivmurashov:~/work/git-extended$ git flow init -f
Which branch should be used for bringing forth production releases?
- develop
- main
Branch name for production releases: [main]

Which branch should be used for integration of the "next release"?
- develop
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/ivmurashov/work/git-extended/.git/hooks]

```

Рис. 4.10: Инициализация git-flow

Проверяю, что я на ветке develop, загружаю весь репозиторий в хранилище (рис. 4.11).

```

ivmurashov@ivmurashov:~/work/git-extended$ git branch
* develop
  main
ivmurashov@ivmurashov:~/work/git-extended$ git push -all
error: имелось в виду '--all' (с двумя дефисами)?
ivmurashov@ivmurashov:~/work/git-extended$ git branch
* develop
  main
ivmurashov@ivmurashov:~/work/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/neve7mind/git-extended/pull/new/develop
remote:
To github.com:neve7mind/git-extended.git
 * [new branch]      develop -> develop

```

Рис. 4.11: Отправка файлов на сервер

Устанавливаю внешнюю ветку как вышестоящую для этой ветки (рис. 4.12).

```

ivmurashov@ivmurashov:~/work/git-extended$ git branch --set-upstream-to=origin/develop develop
branch 'develop' set up to track 'origin/develop'.
ivmurashov@ivmurashov:~/work/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. 4.12: Работа с ветками

Создаю релиз с версией 1.0.0, журнал изменений, добавляю журнал изменений в индекс и заливаю релизную ветку в основную (рис. 4.13).

```

ivmurashov@ivmurashov:~/work/git-extended$ standard-changelog --first-release
✓ output changes to CHANGELOG.md
ivmurashov@ivmurashov:~/work/git-extended$ git add CHANGELOG.md
ivmurashov@ivmurashov:~/work/git-extended$ git commit -am 'chore(site): add changelog'
[develop 254cdad] chore(site): add changelog
1 file changed, 9 deletions(-)
ivmurashov@ivmurashov:~/work/git-extended$ git flow release finish 1.0.0

```

Рис. 4.13: Работа с ветками

Отправляю данные на github (рис. 4.14).

```

ivmurashov@ivmurashov:~/work/git-extended$ git push --all
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (7/7), готово.
Запись объектов: 100% (8/8), 997 байтов | 997.00 КиБ/с, готово.
Всего 8 (изменений 2), повторно использовано 0 (изменений 0), повторно использовано паке
тов 0
remote: Resolving deltas: 100% (2/2), done.
To github.com:neve7mind/git-extended.git
  4bb361c..254cdad develop -> develop
  4bb361c..7f50cbf main -> main
ivmurashov@ivmurashov:~/work/git-extended$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 167 байтов | 167.00 КиБ/с, готово.
Всего 1 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано паке
тов 0
To github.com:neve7mind/git-extended.git
 * [new tag]          v1.0.0 -> v1.0.0

```

Рис. 4.14: Отправка файлов на сервер

Создаю релиз на github (рис. 4.15).

```

ivmurashov@ivmurashov:~/work/git-extended$ gh release create v1.0.0 -F CHANGELOG.md
https://github.com/neve7mind/git-extended/releases/tag/v1.0.0

```

Рис. 4.15: Создание релиза

Создаю ветку для новой функциональности и по окончании разработки но-  
вой функциональности объединяю ветку feature\_branch с develop (рис. 4.16).

```

ivmurashov@ivmurashov:~/work/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

ivmurashov@ivmurashov:~/work/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 254cdad).

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. 4.16: Создание и объединение веток

Создаю релиз с версией 1.2.3 (рис. 4.17).

```

ivmurashov@ivmurashov:~/work/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'

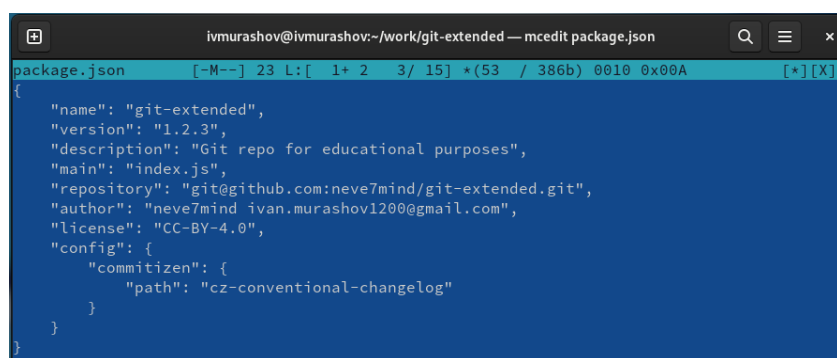
Follow-up actions:
- Bump the version number now!
- Start committing last-minute fixes in preparing your release
- When done, run:

    git flow release finish '1.2.3'

```

Рис. 4.17: Создание релиза

Обновляю номер версии в файле package.json (рис. 4.18).



```

package.json  [-M--] 23 L: [ 1+ 2 3/ 15] *(53 / 386b) 0010 0x00A [*][X]
{
  "name": "git-extended",
  "version": "1.2.3",
  "description": "Git repo for educational purposes",
  "main": "index.js",
  "repository": "git@github.com:neve7mind/git-extended.git",
  "author": "neve7mind ivan.murashov1200@gmail.com",
  "license": "CC-BY-4.0",
  "config": {
    "commitizen": {
      "path": "cz-conventional-changelog"
    }
  }
}

```

Рис. 4.18: Настройка конфигураций

Заливаю релизную ветку в основную, после чего отправляю данные на github и создаю релиз на github с комментарием из журнала изменений (рис. 4.19).

```

2 files changed, 5 insertions(+), 1 deletion(-)
ivmurashov@ivmurashov:~/work/git-extended$ git flow release finish 1.2.3
Переключились на ветку «main»
Эта ветка соответствует «origin/main».
Merge made by the 'ort' strategy.
  CHANGELOG.md | 7 +-----
  package.json | 2 +-
  2 files changed, 2 insertions(+), 7 deletions(-)
Уже на «main»
Ваша ветка опережает «origin/main» на 4 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 55109db).

Summary of actions:
- Release branch 'release/1.2.3' has been merged into 'main'

```

Рис. 4.19: Работа с ветками

## 5 Выводы

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