

# **Отчёт по лабораторной работе №4**

**Продвинутое использование git.**

Жукова Арина Александровна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
3.1	Установка программного обеспечения . . . . .	7
<b>4</b>	<b>Выводы</b>	<b>18</b>
	<b>Список литературы</b>	<b>19</b>

# Список иллюстраций

3.1	Установка git-flow . . . . .	7
3.2	Установка Node.js . . . . .	7
3.3	Использование команды apt-get install pnpm . . . . .	8
3.4	Настройка Node.js . . . . .	8
3.5	Использование команды source ~/.bashrc . . . . .	9
3.6	Формантирование коммитов . . . . .	9
3.7	Создание логов . . . . .	10
3.8	Создание репозитория . . . . .	10
3.9	Создание коммита . . . . .	10
3.10	Выкладывание коммита на github . . . . .	11
3.11	Настройка конфигурации . . . . .	11
3.12	Файл package.json . . . . .	12
3.13	Выполнение команды git add .; git cz; git push . . . . .	12
3.14	Инициализация git-flow . . . . .	12
3.15	Префикс для ярлыков . . . . .	13
3.16	Проверка ветки . . . . .	13
3.17	Выгрузка в хранилище . . . . .	13
3.18	Установка вышестоящей ветки . . . . .	13
3.19	Создание релиза . . . . .	14
3.20	Создание журнала изменений . . . . .	14
3.21	Добавление журнала изменений . . . . .	14
3.22	Соединение релизной ветки и основной . . . . .	14
3.23	Отправка данных на github . . . . .	15
3.24	Отправка данных на github . . . . .	15
3.25	Создание релиза . . . . .	15
3.26	Создание ветки . . . . .	15
3.27	Объединение веток . . . . .	16
3.28	Создание релиза . . . . .	16
3.29	Создание журнала изменений . . . . .	16
3.30	Добавление журнала изменений . . . . .	16
3.31	Соединение релизной ветки и основной . . . . .	17
3.32	Отправка данных на github . . . . .	17
3.33	Отправка данных на github . . . . .	17
3.34	Создание релиза . . . . .	17

## Список таблиц

# 1 Цель работы

Получение навыков правильной работы с репозиториями git.

## 2 Задание

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

## 3 Выполнение лабораторной работы

### 3.1 Установка программного обеспечения

#### 1. Установка git-flow.

Необходимо установить git-flow при помощи команд: `dnf copr enable elegos/gitflow` и `dnf install gitflow` (рис. 3.1).

```
root@aazhukoval:~# dnf install gitflow
Copr repo for gitflow owned by elegos                               3.4 kB/s | 1.5 kB
Последняя проверка окончания срока действия метаданных: 0:00:01 назад, Сб 09 мар 2024 15:49:39.
Зависимости разрешены.
=====
Пакет      Архитектура  Версия      Репозиторий
-----
Установка:
gitflow    x86_64       1.12.3-1.fc34  copr:copr.fedorainfracloud.org:elegos:gitflow
Результат транзакции
=====
```

Рис. 3.1: Установка git-flow

#### 2. Установка Node.js с помощью команд: `dnf install nodejs` и `apt-get install npm` (рис. 3.2-3.3).

```
root@aazhukoval:~# dnf install nodejs
Последняя проверка окончания срока действия метаданных: 0:00:56 назад, Сб 09 мар 2024 15:49:39.
Зависимости разрешены.
=====
Пакет      Архитектура  Версия      Репозиторий
-----
Установка:
nodejs     x86_64       1:20.10.0-3.fc39  updates
Установка зависимостей:
nodejs-libs x86_64       1:20.10.0-3.fc39  updates
Установка слабых зависимостей:
```

Рис. 3.2: Установка Node.js

```
root@aazhukoval:~# apt-get install pnpm
Чтение списков пакетов... Готово
```

Рис. 3.3: Использование команды apt-get install pnpm

### 3. Настройка Node.js

Для дальнейшей работы нам необходимо запустить pnpm при помощи команды pnpm setup (рис. 3.4).

```
root@aazhukoval:~# pnpm setup
Appended new lines to /root/.bashrc

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

To start using pnpm, run:
source /root/.bashrc
```

Рис. 3.4: Настройка Node.js

Выполняем команду: source ~/.bashrc (рис. [-fig. 3.5).



```

root@aazhukoval:~# source ~/.bashrc
root@aazhukoval:~# pnpm add -g commitizen
Packages: +152
+-----+
Downloading registry.npmjs.org/typescript/5.4.2: 5,82 MB/5,82 MB, d
Progress: resolved 152, reused 0, downloaded 152, added 152, done

/root/.local/share/pnpm/global/5:
+ commitizen 4.3.0

Done in 26.5s

```

Рис. 3.5: Использование команды source ~/.bashrc

## 4. Общепринятые коммиты

### 4.1 commitizen

Для помощи в форматировании коммитов используем `pnpm add -g commitizen` (рис. 3.6).

```

root@aazhukoval:~# source ~/.bashrc
root@aazhukoval:~# pnpm add -g commitizen
Packages: +152
+-----+
Downloading registry.npmjs.org/typescript/5.4.2: 5,82 MB/5,82 MB, d
Progress: resolved 152, reused 0, downloaded 152, added 152, done

/root/.local/share/pnpm/global/5:
+ commitizen 4.3.0

Done in 26.5s

```

Рис. 3.6: Формантирование коммитов

### 4.2 standard-changelog

Для помощи в создании логов используем `pnpm add -g standard-changelog` (рис. 3.7).

```
root@aazhukova1:~# pnpm add -g standard-changelog
Packages: +56
Progress: resolved 208, reused 152, downloaded 56, added
/root/.local/share/pnpm/global/5:
+ standard-changelog 5.0.0
Done in 11.9s
```

Рис. 3.7: Создание логов

## 5. Практический сценарий использования git

### 5.1 Создание репозитория git

Создаём репозиторий на GitHub, называем его git-extended. (рис. 3.8).

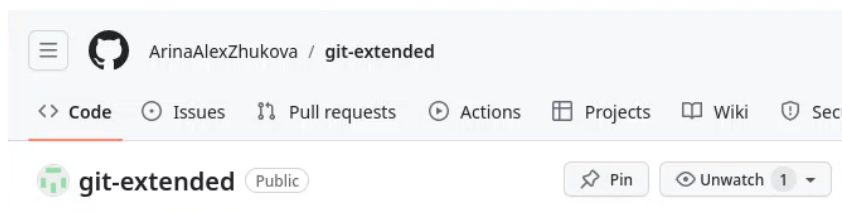


Рис. 3.8: Создание репозитория

### 5.2 Делаем первый коммит и выкладываем на github.

Используем команды: `git commit -m "first commit"`, `git remote add origin git@github.com:ArinaAlexZhukova/git-extended.git` (рис. 3.9).

```
root@aazhukova1:~/git-extended# git commit -m "first commit"
Текущая ветка: main
Эта ветка соответствует «origin/main».

ничего коммитить, нет изменений в рабочем каталоге
root@aazhukova1:~/git-extended# git remote add origin git@github.com:ArinaAlexZhukova/git-extended.git
error: внешний репозиторий origin уже существует
```

Рис. 3.9: Создание коммита

`git push -u origin master` (рис. 3.10).

```
root@aazhukoval:~/git-extended# git push -u origin main
branch 'main' set up to track 'origin/main'.
Everything up-to-date
```

Рис. 3.10: Выкладывание коммита на github

### 5.3 Конфигурация общепринятых коммитов

Настраиваем конфигурация для пакетов Node.js при помощи команды: `pnpm init` (рис. 3.11).

```
root@aazhukoval:~/git-extended# pnpm init
Wrote to /root/git-extended/package.json

{
  "name": "git-extended",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  }
}
```

Рис. 3.11: Настройка конфигурации

Заполняем некоторые параметры пакета. Сконфигурируем формат коммитов, добавляя для в файл `package.json` команду для формирования коммитов (рис. 3.12).

```

package.json  [-M--] 67 L:[ 1+ 5 6/ 16] *(193 / 374b) 0010 0x0
{
  "name": "git-extended",
  "version": "1.0.0",
  "description": "Git repo for educational purposes",
  "main": "index.js",
  "repository": "git@github.com:ArinaAlexZhukova/git-extended.git",
  "keywords": [],
  "author": "Arina Zhukova 1132239120@pfur.ru",
  "license": "CC-BY-4.0",
  "config": {
    "commitizen": {
      "path": "cz-conventional-changelog"
    }
  }
}

```

Рис. 3.12: Файл package.json

При помощи команды `git add .` добавляем новые файлы. Выполняем коммит при помощи команды `git cz`, а также выгружаем на github (рис. 3.13).

```

root@aaazhukoval:~/git-extended# git add .
root@aaazhukoval:~/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) readme.md
? Write a short, imperative tense description of the change (max 83 chars):
  (10) added file
? 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 5627b27] feat(readme.md): added file
1 file changed, 15 insertions(+)
create mode 100644 package.json
root@aaazhukoval:~/git-extended# git push
Перечисление объектов: 4, готово.
Подсчет объектов: 100% (4/4), готово.
Сжатие объектов: 100% (3/3), готово.

```

Рис. 3.13: Выполнение команды `git add .`; `git cz`; `git push`

## 5.4 Конфигурация git-flow

Инициализируем git-flow при помощи команды `git flow init` (рис. 3.14).

```

root@aaazhukoval:~/git-extended# git flow init

Which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main]
Branch name for "next release" development: [develop]

```

Рис. 3.14: Инициализация git-flow

Устанавливаем префикс для ярлыков в v 3.15).

```
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? [/root/git-extended/.git/hooks]
```

Рис. 3.15: Префикс для ярлыков

Проверяем, что находимся на ветке branch (рис. 3.16).

```
root@aazhukoval:~/git-extended# git branch
* develop
main
```

Рис. 3.16: Проверка ветки

Загружаем весь репозиторий в хранилище при помощи `git push --all` (рис. 3.17).

```
root@aazhukoval:~/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/ArinaAlexZhukova/git-extended/pull/new/develop
remote:
To github.com:ArinaAlexZhukova/git-extended.git
 * [new branch]      develop -> develop
```

Рис. 3.17: Выгрузка в хранилище

Устанавливаем внешнюю ветку как вышестоящую для этой ветки (рис. 3.18).

```
root@aazhukoval:~/git-extended# git branch --set-upstream-to=origin/develop develop
branch 'develop' set up to track 'origin/develop'.
```

Рис. 3.18: Установка вышестоящей ветки

Создание релиза с версией 1.0.0 при помощи `git flow release start 1.0.0` (рис. 3.19).

```

root@aazhukoval:~/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. 3.19: Создание релиза

Создание журнала изменений с помощью команды `standard-changelog --first-release` (рис. 3.20).

```

root@aazhukoval:~/git-extended# standard-changelog --first-release
✓ created CHANGELOG.md
✓ output changes to CHANGELOG.md

```

Рис. 3.20: Создание журнала изменений

Добавление журнала изменений в индекс (рис. 3.21).

```

root@aazhukoval:~/git-extended# git add CHANGELOG.md
root@aazhukoval:~/git-extended# git commit -am 'chore(site): add changelog'
[release/1.0.0 8ac06f5] chore(site): add changelog
1 file changed, 9 insertions(+)
create mode 100644 CHANGELOG.md

```

Рис. 3.21: Добавление журнала изменений

Зальём релизную ветку в основную ветку (рис. 3.22).

```

root@aazhukoval:~/git-extended# git flow release finish 1.0.0
Branches 'main' and 'origin/main' have diverged.
And local branch 'main' is ahead of 'origin/main'.
Уже на «main»
Ваша ветка опережает «origin/main» 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 ++++++++

```

Рис. 3.22: Соединение релизной ветки и основной

Отправка данных на github (рис. 3.23-3.24).

```

root@aazhukoval:~/git-extended# git push --all
Перечисление объектов: 6, готово.
Подсчет объектов: 100% (6/6), готово.
Сжатие объектов: 100% (5/5), готово.
Запись объектов: 100% (5/5), 2.81 КиБ | 1.41 МиБ/с, готово.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:ArinaAlexZhukova/git-extended.git
   5627b27..2ff4750  develop -> develop
   5627b27..75a330c  main -> main
root@aazhukoval:~/git-extended# git push --tags

```

Рис. 3.23: Отправка данных на github

```

5627b27..75a330c  main -> main
root@aazhukoval:~/git-extended# git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 160 байтов | 160.00 КиБ/с, готово.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:ArinaAlexZhukova/git-extended.git
 * [new tag]          v1.0.0 -> v1.0.0

```

Рис. 3.24: Отправка данных на github

Создание релиза на github (рис. 3.25).

```

root@aazhukoval:~/git-extended# gh release create v1.0.0 -F CHANGELOG.md
https://github.com/ArinaAlexZhukova/git-extended/releases/tag/v1.0.0

```

Рис. 3.25: Создание релиза

## 6. Работа с репозиторием git

### 6.1 Разработка новой функциональности

Создание ветки для новой функциональности (рис. 3.26).

```

root@aazhukoval:~/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. 3.26: Создание ветки

Объединение ветки feature\_branch с develop (рис. 3.27).

```

root@aazhukoval:~/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 2ff4750).

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. 3.27: Объединение веток

## 6.2 Создание релиза git-flow

Создаём релиз с версией 1.2.3 (рис. 3.28).

```

root@aazhukoval:~/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. 3.28: Создание релиза

Обновляем номер версии в файле package.json. Создаём журнал изменений (рис. 3.29).

```

root@aazhukoval:~/git-extended# standard-changelog
✓ output changes to CHANGELOG.md
root@aazhukoval:~/git-extended# git add CHANGELOG.md

```

Рис. 3.29: Создание журнала изменений

Добавим журнал изменений в индекс (рис. 3.30).

```

root@aazhukoval:~/git-extended# git add CHANGELOG.md
root@aazhukoval:~/git-extended# git commit -am 'chore(site): update changelog'
[release/1.2.3 bf5e832] chore(site): update changelog
2 files changed, 5 insertions(+), 1 deletion(-)

```

Рис. 3.30: Добавление журнала изменений



Зальём релизную ветку в основную ветку (рис. 3.31).

```
root@aaazhukoval:~/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 | 4 ++++
  package.json | 2 +-
  2 files changed, 5 insertions(+), 1 deletion(-)
Уже на «main»
Ваша ветка опережает «origin/main» на 3 коммита.
(используйте «git push», чтобы опубликовать ваши локальные коммиты)
Переключились на ветку «develop»
```

Рис. 3.31: Соединение релизной ветки и основной

Отправка данных на github (рис. 3.32-3.33).

```
root@aaazhukoval:~/git-extended# git push --all
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 2.78 КиБ | 2.78 МБ/с, готово.
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 github.com:ArinaAlexZhukova/git-extended.git
  2ff4750..77a3eb9  develop -> develop
  75a330c..c578ae5  main -> main
```

Рис. 3.32: Отправка данных на github

```
root@aaazhukoval:~/git-extended# git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 160 байтов | 1024 байта/с, готово.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:ArinaAlexZhukova/git-extended.git
 * [new tag]          v1.2.3 -> v1.2.3
```

Рис. 3.33: Отправка данных на github

Создание релиза на github с комментарием из журнала изменений (рис. 3.34).

```
root@aaazhukoval:~/git-extended# gh release create v1.2.3 -F CHANGELOG.md
https://github.com/ArinaAlexZhukova/git-extended/releases/tag/v1.2.3
```

Рис. 3.34: Создание релиза

## 4 Выводы

Нами были изучены и опробованы на практике многие возможности git-flow.

## **Список литературы**