

# Лабораторная работа № 4

## Продвинутое использование git

---

Жукова С. В. НПИбд-01-24

5 марта 2024

Российский университет дружбы народов, Москва, Россия

- Жукова София Викторовна
- студентка
- направления прикладной информатика
- Российский университет дружбы народов
- 1032240966@pfur.ru
- <https://svzhukova.github.io/ru/>



## Вводная часть

---

## Цель работы

Получение навыков правильной работы с репозиториями git.

## Задание

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

## Выполнение лабораторной работы

---

# Установка программного обеспечения

## Установим git-flow

```
[svzhukova@svzhukova ~]$ sudo -i  
[sudo] пароль для svzhukova:
```

Figure 1: Установим git-flow

```
[root@svzhukova ~]# 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 реп  
озитория.  
  
Do you really want to enable copr.fedorainfracloud.org/elegos/gitflow? [y/  
N]: y  
Репозиторий успешно подключен.
```

Figure 2: # Enable the copr repository

```
[root@svzhukova ~]# dnf install gitflow
Copr repo for gitflow owned by elegos      1.1 kB/s | 919 B    00:00
Fedora 40 - aarch64 - Updates              34 kB/s | 18 kB    00:00
Fedora 40 - aarch64 [=====] 59 kB/s | 18 kB    00:00 ETA
■
```

Figure 3: # Install gitflow

# Установка Node.js

На Node.js базируется программное обеспечение для семантического версионирования и общепринятых коммитов.

```
[root@svzhukova ~]# dnf install nodejs
Последняя проверка окончания срока действия метаданных: 0:02:59 назад, Пн
03 мар 2025 15:49:56.
Зависимости разрешены.
=====
 Пакет          Архитектура      Версия      Репозиторий      Размер
=====
 Установка:
 nodejs        aarch64    1:20.18.2-2.fc40      updates      53 k
 Установка зависимостей:
 nodejs-langs   aarch64    1:20.18.2-2.fc40      updates      15 M
 Установка слабых зависимостей:
 nodejs-docs    noarch     1:20.18.2-2.fc40      updates      8.4 M
 nodejs-full-i18n aarch64    1:20.18.2-2.fc40      updates      8.4 M
 nodejs-npm     aarch64    1:10.8.2-1.20.18.2.2.fc40  updates      2.1 M
 Результат транзакции
=====
 Установка 5 Пакетов

Объем загрузки: 34 M
Объем изменений: 186 M
Продолжить? [д/н]: у
Загрузка пакетов:
(1/5): nodejs-20.18.2-2.fc40.aarch64.rpm: 415 kB/s | 53 kB     00:00
[2-4/5]: nodejs-1.70% [=====----] 7.0 MB/s | 17 MB     00:02 ETA
```

# Настройка Node.js

Для работы с Node.js добавим каталог с исполняемыми файлами, устанавливаемыми yarn, в переменную PATH.

```
Выполнено!
[root@svzhukova ~]# dnf install rpm
Последняя проверка окончания срока действия метаданных: 0:03:56 назад, Пн
03 мар 2025 15:49:56.
Зависимости разрешены.
=====
Пакет      Архитектура      Версия      Репозиторий      Размер
=====
Установка:
rpm        noarch        9.0.6-1.fc40    updates        2.8 M

Результат транзакции
=====
Установка 1 Пакет

Объем загрузки: 2.8 М
Объем изменений: 14 М
Продолжить? [д/н]: у
Загрузка пакетов:
rpm-9.0.6-1.fc40.noarch.rpm          7.6 MB/s | 2.8 MB   00:00

Общий размер                      3.2 MB/s | 2.8 MB   00:00
Проверка транзакции
Проверка транзакции успешно завершена.
Идет проверка транзакции
Тест транзакции проведен успешно.
Выполнение транзакции
```



## Выполним

```
[root@svzhukova ~]# 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
[root@svzhukova ~]#
```

Figure 6: Установка

commitizen

Данная программа используется для помощи в форматировании  
коммитов

```
[root@svzhukova ~]# source ~/.bashrc
[root@svzhukova ~]# █
```

Figure 7: Форматирование коммитов

Данная программа используется для помощи в создании логов.

```
[root@svzhukova ~]# pnpm add -g commitizen

          Update available! 9.0.6 → 10.5.2.
Changelog: https://github.com/pnpm/pnpm/releases/tag/v10.5.2
Run "pnpm add -g pnpm" to update.

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

/root/.local/share/pnpm/global/5:
+ commitizen 4.3.1
```

Figure 8: Форматирование коммитов

# Практический сценарий использования git

---

# Создание репозитория git

## Подключение репозитория к github

Создайте репозиторий на GitHub. Для примера назовём его git-extended.

```
[root@svzhukova ~]# 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

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

Done in 2.5s
[root@svzhukova ~]# █
```

Figure 9: Подключение репозитория к github

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

*Required fields are marked with an asterisk (\*).*

**Repository template**

No template ▾

Start your repository with a template repository's contents.

**Owner \***      **Repository name \***

 svzhukova /  git-extended is available.

Great repository names are short and memorable. Need inspiration? How about [didactic-octo-enigma](#) ?

**Description (optional)**

 Public  
Anyone on the internet can see this repository. You choose who can commit.

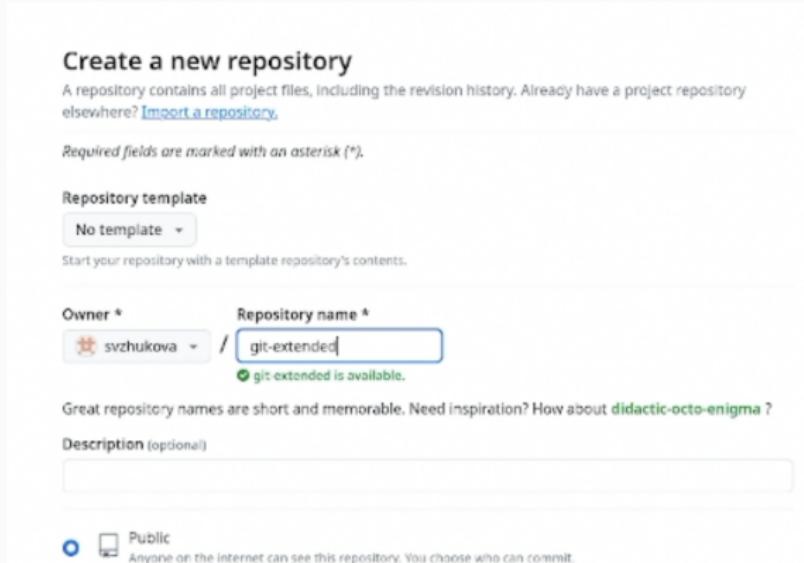


Figure 10: Подключение репозитория к github

# Делаем первый коммит и выкладываем на github

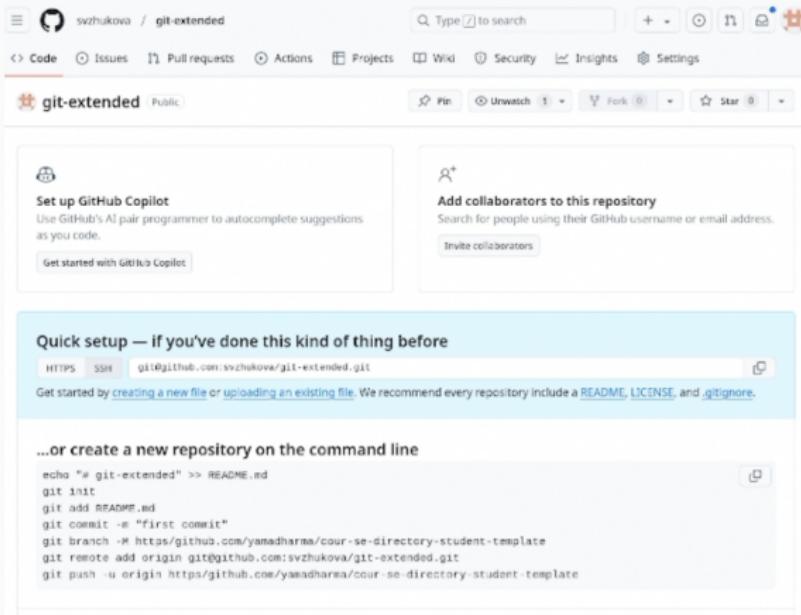


Figure 11: Делаем первый коммит

## Конфигурация общепринятых коммитов

### Конфигурация для пакетов Node.js

```
[svzhukova@svzhukova ~]$ git clone --recursive git@gith  
ub.com:svzhukova/git-extended.git  
Клонирование в «git-extended»...  
warning: Похоже, что вы клонировали пустой репозиторий.  
[svzhukova@svzhukova ~]$ █
```

Figure 12: Конфигурация для пакетов Node.js

Необходимо заполнить несколько параметров пакета.

```
[svzhukova@svzhukova ~]$ cd git-extended  
[svzhukova@svzhukova git-extended]$ git commit -m "fir  
st commit"  
Текущая ветка: https://github.com/yamadharma/cour  
se-directory-student-template  
  
Начальный коммит  
  
чего коммитить (создайте/скопируйте файлы, затем запу  
стите  
«git add», чтобы отслеживать их)  
[svzhukova@svzhukova git-extended]$ █
```

Figure 13: Сконфигурируем формат коммитов

Добавим новые файлы, Выполним коммит, Отправим на github

```
[svzhukova@svzhukova git-extended]$ git remote add orig  
in git@github.com:svzhukova/git-extended.git  
error: внешний репозиторий origin уже существует  
[svzhukova@svzhukova git-extended]$ █
```

Figure 14: Добавим новые файлы

# Конфигурация git-flow

Инициализируем git-flow

```
[svzhukova@svzhukova git-extended]$ git push -u origin
```

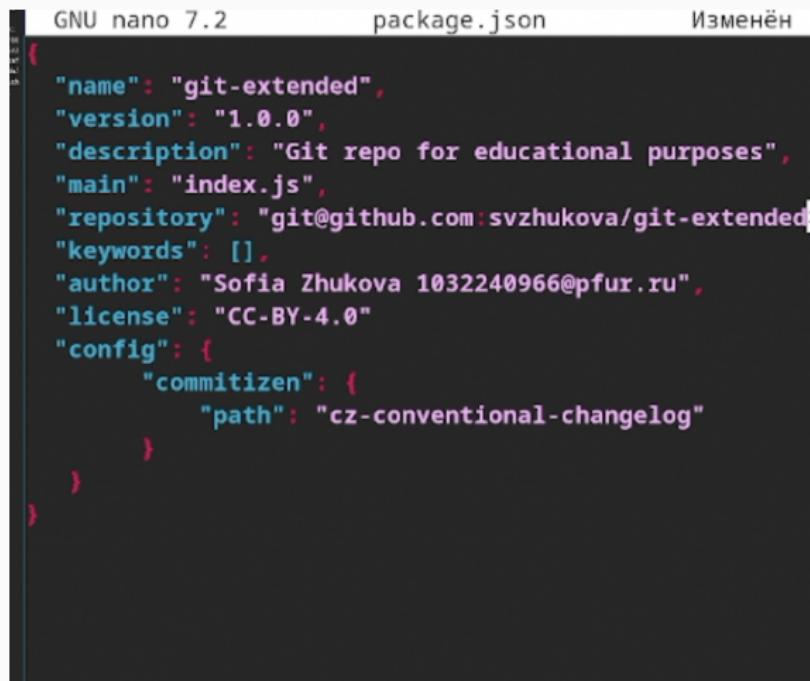
Figure 15: Инициализируем git-flow

Предфикс для ярлыков установим в V.

```
[svzhukova@svzhukova git-extended]$ pnpm init
Wrote to /home/svzhukova/git-extended/package.json

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

## Загрузим весь репозиторий в хранилище



The screenshot shows a terminal window with three tabs: 'GNU nano 7.2', 'package.json', and 'Изменён'. The 'package.json' tab is active, displaying the following JSON code:

```
{  
  "name": "git-extended",  
  "version": "1.0.0",  
  "description": "Git repo for educational purposes",  
  "main": "index.js",  
  "repository": "git@github.com:svzhukova/git-extended",  
  "keywords": [],  
  "author": "Sofia Zhukova 1032240966@pfur.ru",  
  "license": "CC-BY-4.0",  
  "config": {  
    "commitizen": {  
      "path": "cz-conventional-changelog"  
    }  
  }  
}
```

Figure 17: Загрузим весь репозиторий в хранилище

Установим внешнюю ветку как вышестоящую для этой ветки

```
[svzhukova@svzhukova git-extended]$ git add .
```

Figure 18: Установим внешнюю ветку как вышестоящую для этой ветки

Создадим релиз с версией 1.0.0

```
[svzhukova@svzhukova git-extended]$ git commit -am "feat(main): make course structure"
```

Figure 19: Создадим релиз с версией 1.0.0

## Создадим журнал изменений

```
[svzhukova@svzhukova 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, г
отово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (f
rom 0)
To github.com:svzhukova/git-extended.git
  df0bae8..835ce08  https://github.com/yamadharma/cour-s
e-directory-student-template -> https://github.com/yamadh
arma/cour-se-directory-student-template
```

Figure 20: Создадим журнал изменений

Добавим журнал изменений в индекс

```
[svzhukova@svzhukova git-extended]$ git flow init
Which branch should be used for bringing forth producti
on releases?
  - https://github.com/yamadharma/cour-se-directory-stud
ent-template
Branch name for production releases: [
```

## Зальём релизную ветку в основную ветку

```
[svzhukova@svzhukova git-extended]$ git branch
* develop
  https://github.com/yamadharma/course-directory-student
-template
[svzhukova@svzhukova git-extended]$
```

Figure 22: Зальём релизную ветку в основную ветку

Отправим данные на github

```
[svzhukova@svzhukova git-extended]$ git push --all
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (f
rom 0)
remote:
remote: Create a pull request for 'develop' on GitHub b
y visiting:
remote:     https://github.com/svzhukova/git-extended/
pull/new/develop
remote:
To github.com:svzhukova/git-extended.git
 * [new branch]      develop -> develop
```

Figure 23: Отправим данные на github

Создадим релиз на github. Для этого будем использовать утилиты работы с github

```
[svzhukova@svzhukova git-extended]$ git branch --set-up  
stream-to=origin/develop develop  
branch 'develop' set up to track 'origin/develop'.  
[svzhukova@svzhukova git-extended]$
```

Figure 24: Создадим релиз на github

# Работа с репозиторием git

## Разработка новой функциональности

```
[svzhukova@svzhukova 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'
```

Figure 25: Создадим ветку для новой функциональности

Далее, продолжаем работу с git как обычно.

```
[svzhukova@svzhukova git-extended]$ standa  
rd-changelog --version  
6.0.0  
[svzhukova@svzhukova git-extended]$ standa  
rd-changelog --first-release  
✓ created CHANGELOG.md  
✓ output changes to CHANGELOG.md
```

Figure 26: Объединим ветку feature\_branch с develop

## Создание релиза git-flow

```
[svzhukova@svzhukova git-extended]$ git add CHANGELOG.md
[svzhukova@svzhukova git-extended]$ git commit -am 'chore(site): add changelog'
[release/1.0.0 10eb7f6] chore(site): add changelog
 1 file changed, 9 insertions(+)
 create mode 100644 CHANGELOG.md
```

Figure 27: Создадим релиз с версией 1.2.3

## Установим её в 1.2.3.

Создадим журнал изменений

```
[svzhukova@svzhukova git-extended]$ git commit -m "v1.2.3: finish 1.0.0"
```

Figure 28: Обновим номер версии в файле package.json

Добавим журнал изменений в индекс

```
[svzhukova@svzhukova git-extended]$ git push --all
sh: --all: command not found
Перечисление объектов: 6, готово.
Подсчет объектов: 100% (6/6), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (5/5), готово.
Запись объектов: 100% (5/5), 725 байтов | 725.00 КиБ/с, готово.
Total 5 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), done
.
To github.com:svzhukova/git-extended.git
  835ce08..ea004e6  develop -> develop
  835ce08..65e8538  https://github.com/yama
```

```
[svzhukova@svzhukova git-extended]$ gh release create v1.0.0 -F CHANGELOG.md
```

Figure 30: Зальём релизную ветку в основную ветку

Отправим данные на github

```
[svzhukova@svzhukova 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
```

## Создадим релиз на github с комментарием из журнала изменений

```
[svzhukova@svzhukova 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
ea004e6).

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'
```

Figure 32: Создадим релиз на github

## Заключение

Мы получили навыки правильной работы с репозиториями git.