

# Лабораторная работа №4

Продвинутое использование git

---

Комягин А.Н.

9 марта 2024

Российский университет дружбы народов, Москва, Россия

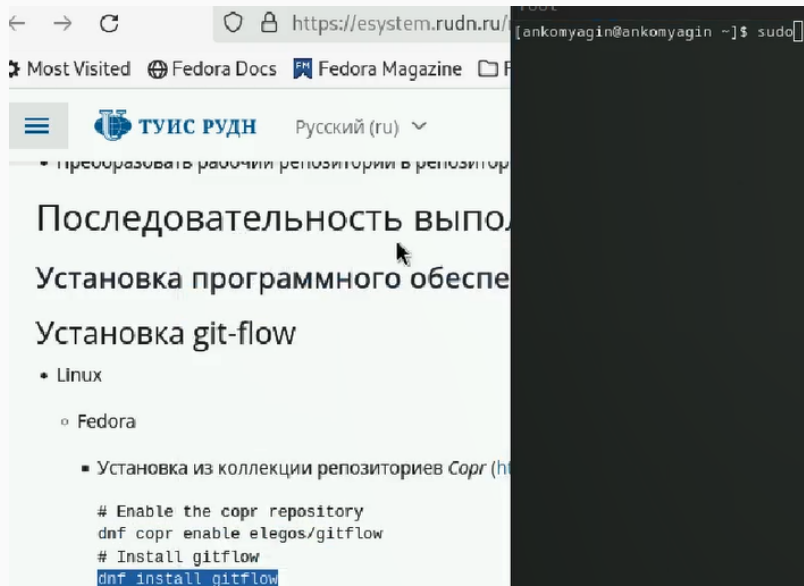
Получение навыков продвинутой работы с репозиториями git.

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

# Выполнение лабораторной работы


---

# Установка необходимого ПО



The screenshot shows a web browser window with the URL `https://esystem.rudn.ru/`. The page header includes the TUIS RUDN logo and a language selector set to Russian (ru). The main content area displays the title "Последовательность выполнения" (Sequence of execution) and the subtitle "Установка программного обеспечения" (Installation of software). Below this, the section "Установка git-flow" (Installation of git-flow) is shown, with a list of operating systems: Linux and Fedora. Under the Fedora section, there is a link to "Установка из коллекции репозитория Copr (https://copr.fedoraproject.org/coprs/elegos/gitflow/)" and a list of terminal commands to install git-flow from the Copr repository. The command `dnf install gitflow` is highlighted in blue.

← → ↺ `https://esystem.rudn.ru/` Most Visited Fedora Docs Fedora Magazine

☰  **ТУИС РУДН** Русский (ru) ▾

• преобразовать рабочий репозиторий в репозиторий

## Последовательность выполнения

### Установка программного обеспечения

#### Установка git-flow

- Linux
  - Fedora
    - Установка из коллекции репозитория Copr (<https://copr.fedoraproject.org/coprs/elegos/gitflow/>)

```
# Enable the copr repository
dnf copr enable elegos/gitflow
# Install gitflow
dnf install gitflow
```

## Настройка Node.js

Для работы с Node.js добавим каталог с исполняемыми файлами.

- Запустите:

```
pnpm setup
```

- Перелогиньтесь, или выполните:

```
source ~/.bashrc
```

```
Построение дерева зависимостей... Готово
E: Невозможно найти пакет pnpm
[ankomyagin@ankomyagin ~]$ wget -qO- https://get.pnpm.io/install.sh | sh
==> Downloading pnpm binaries 8.15.4

WARN using --force I sure hope you know what you are doing
Copying pnpm CLI from /tmp/tmp.1Xdm5h0Drg/pnpm to /home/ankomyagin/.local/share/pnpm/pnpm
Appended new lines to /home/ankomyagin/.bashrc

Next configuration changes were made:
export PNPM_HOME="/home/ankomyagin/.local/share/pnpm"
```

# Создание репозитория

```
[ankomyagin@ankomyagin ~]$ cd work/
[ankomyagin@ankomyagin work]$ git clone --recurse git@github.com:ANKomyagin/git-extended.git
Клонирование в «git-extended»...
warning: Похоже, что вы клонировали пустой репозиторий.
[ankomyagin@ankomyagin work]$ cd
ANKomyagin.github.io/ git-extended/
blog/          study/
[ankomyagin@ankomyagin work]$ cd git-extended/
[ankomyagin@ankomyagin git-extended]$ touch text.txt
[ankomyagin@ankomyagin git-extended]$ git commit -m "first commit"
Текущая ветка: main

Начальный коммит

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

индекс пуст, но есть неотслеживаемые файлы
(используйте «git add», чтобы проиндексировать их)
[ankomyagin@ankomyagin git-extended]$ git add .
[ankomyagin@ankomyagin git-extended]$ git commit -m "first commit"
[main (корневой коммит) 5c567c7] first commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 text.txt
[ankomyagin@ankomyagin git-extended]$
```

# Создание репозитория

```
[ankomyagin@ankomyagin git-extended]$ git remote add origin git@github.com:ANKomyagin/git-extended.git
error: внешний репозиторий origin уже существует
[ankomyagin@ankomyagin git-extended]$ git push -u origin master
error: src refspec master ничему не соответствует
error: не удалось отправить некоторые ссылки в «github.com:ANKomyagin/git-extended.git»
[ankomyagin@ankomyagin git-extended]$ git push -u origin main
Перечисление объектов: 3, готово.
Подсчет объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 866 байтов | 866.00 КиБ/с, готово.
Всего 3 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
To github.com:ANKomyagin/git-extended.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
[ankomyagin@ankomyagin git-extended]$
```

Рис. 2: ed25519



## Изменяем параметры пакета

```
(  
  "name": "git-extended",  
  "version": "1.0.0",  
  "description": "Git repo for educational purposes",  
  "main": "index.js",  
  "repository": "git@github.com:username/git-extended.git",  
  "author": "Andrey Komyagin Komyagin12345@mail.ru",  
  "license": "CC-BY-4.0",  
  "config": {  
    "commitizen": {  
      "path": "cz-conventional-changelog"  
    }  
  }  
),
```

```
[ankomyagin@ankomyagin git-extended]$ nano package.json
[ankomyagin@ankomyagin git-extended]$ git add .
[ankomyagin@ankomyagin 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)

## Инициализация git-flow

```
[ankomyagin@ankomyagin 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]

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
```

## операция над веткой

- Установите внешнюю ветку как вышестоящую для этой ветки:

```
git branch --set-upstream-to=origin/develop develop
```

- Создадим релиз с версией 1.0.0

```
git flow release start 1.0.0
```

- Создадим журнал изменений

```
standard-changelog --first-release
```

- Добавим журнал изменений в индекс

```
git add CHANGELOG.md
```

```
git commit -am 'chore(site): add changelog'
```

- Создадим релиз с версией 1.2.3:

```
git flow release start 1.2.3
```

- Обновите номер версии в файле `package.json`. Установите её в 1.2.3.
- Создадим журнал изменений

```
standard-changelog
```

- Добавим журнал изменений в индекс

```
git add CHANGELOG.md  
git commit -am 'chore(site): update changelog'
```

## создадим релиз изменений

```
Подсчет объектов: 100% (1/1), готово.  
Запись объектов: 100% (1/1), 160 байтов | 160.00 КиБ/с, готово.  
Всего 1 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано  
кетов 0  
To github.com:ANKomyagin/git-extended.git  
* [new tag]          v1.2.3 -> v1.2.3  
[ankomyagin@ankomyagin git-extended]$ gh release create v1.2.3 -F CHANGELOG.md
```

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