

# **Отчёт по лабораторной работе №4**

**Продвинутое использование git**

Поляков Сергей Александрович

# **Содержание**

<b>1 Цель работы</b>	<b>5</b>
<b>2 Теоретические сведения</b>	<b>6</b>
<b>3 Выполнение лабораторной работы</b>	<b>8</b>
3.1 Работа с тестовым репозиторием . . . . .	8
3.2 Подготовка рабочего репозитория . . . . .	19
<b>4 Вывод</b>	<b>21</b>

# Список иллюстраций

3.1 Node.js . . . . .	9
3.2 установка commitizen . . . . .	10
3.3 установка standard-changelog . . . . .	11
3.4 package.json . . . . .	12
3.5 Отправка . . . . .	13
3.6 Инициализация и загрузка изменений . . . . .	14
3.7 Завершение релиза . . . . .	15
3.8 Отправка . . . . .	16
3.9 Объединение веток . . . . .	17
3.10 Завершение релиза . . . . .	18
3.11 package.json и коммит . . . . .	19
3.12 Завершение релиза . . . . .	20

# **Список таблиц**

# **1 Цель работы**

Получение навыков правильной работы с репозиториями git.

## 2 Теоретические сведения

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

- Когда работа над веткой исправления hotfix завершена, она сливаются в ветки develop и master.

# **3 Выполнение лабораторной работы**

## **3.1 Работа с тестовым репозиторием**

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

```
sapolyakov@sapolyakov:~$  
sapolyakov@sapolyakov:~$ pnpm setup  
Appended new lines to /home/sapolyakov/.bashrc  
  
Next configuration changes were made:  
export PNPM_HOME="/home/sapolyakov/.local/share/pnpm"  
case ":$PATH:" in  
  *":$PNPM_HOME:"*) ;;  
  *) export PATH="$PNPM_HOME:$PATH" ;;  
esac  
  
To start using pnpm, run:  
source /home/sapolyakov/.bashrc  
sapolyakov@sapolyakov:~$ source ~/.bashrc  
sapolyakov@sapolyakov:~$
```

Рисунок 3.1: Node.js

Программа commitizen используется для помощи в форматировании коммитов. При этом устанавливается скрипт git-cz, который мы и будем использовать для коммитов.

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

Update available! 10.27.0 → 10.30.0.
Changelog: https://pnpm.io/v/10.30.0
To update, run: pnpm self-update

[

WARN 2 deprecated subdependencies found: glob@7.2.3, inflight@1.0.6
Packages: +149
+++++
Progress: resolved 149, reused 0, downloaded 149, added 149, done

/home/sapolyakov/.local/share/pnpm/global/5:
+ commitizen 4.3.1

Done in 2.5s using pnpm v10.27.0
sapolyakov@sapolyakov:~$
```

Рисунок 3.2: установка commitizen

Программа standard-changelog используется для помощи в создании логов.

```
sapolyakov@sapolyakov:~$ pnpm add -g standard-changelog
WARN 2 deprecated subdependencies found: glob@7.2.3, inflight@1.0.6
Packages: +32
+++++
Progress: resolved 181, reused 149, downloaded 32, added 32, done

/home/sapolyakov/.local/share/pnpm/global/5:
+ standard-changelog 7.0.1

Done in 2s using pnpm v10.27.0
sapolyakov@sapolyakov:~$
```

Рисунок 3.3: установка standard-changelog

Делаем первый коммит и выкладываем на github.

Необходимо заполнить несколько параметров пакета.

Таким образом, файл package.json приобретает вид:



The screenshot shows a code editor window with the following interface elements:

- Top bar: "Open" button, dropdown menu, "+" button, file name "package.json", path "~/git-extended", and "Save" button.
- Code area: A JSON object with numbered lines 1 through 15. Lines 1 through 14 are visible, and line 15 has a cursor.
- Bottom status bar: Shows a progress bar with two segments: a grey segment and a white segment.

```
1{
2  "name": "git-extended",
3  "version": "1.0.0",
4  "description": "Git repo for educational purposes",
5  "main": "index.js",
6  "repository": "git@github.com:sapolyakov535/git-extended.git",
7  "author": "sapolyakov535",
8  "license": "CC-BY-4.0",
9  "config": {
10    "commitizen": {
11      "path": "cz-conventional-changelog"
12    }
13  }
14}
15|
```

Рисунок 3.4: package.json

Добавим новые файлы.

Выполним коммит.

Отправим на github.

```
sapolyakov@sapolyakov:~/git-extended$ git add .
sapolyakov@sapolyakov:~/git-extended$ git cz
cz-cli@4.3.1, cz-conventional-changelog@3.3.0

? Select the type of change that you're committing: chore: Other changes that don't modify src or test files
? What is the scope of this change (e.g. component or file name): (press enter to skip)
? Write a short, imperative tense description of the change (max 93 chars):
(4) test
? 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
[master f8f7241] chore: test
  1 file changed, 15 insertions(+)
  create mode 100644 package.json
sapolyakov@sapolyakov:~/git-extended$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 1.12 KiB | 1.12 MiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:sapolyakov535/git-extended.git
  ee3098b..f8f7241 master -> master
sapolyakov@sapolyakov:~/git-extended$
```

Рисунок 3.5: Отправка

Инициализируем git-flow

Проверьте, что Вы на ветке develop

Загрузите весь репозиторий в хранилище

```
sapolyakov@sapolyakov:~/git-extended$ git flow init

Which branch should be used for bringing forth production releases?
  - master
Branch name for production releases: [master]
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/sapolyakov/git-extended/.git/hooks]
sapolyakov@sapolyakov:~/git-extended$ git branch
* develop
  master
sapolyakov@sapolyakov:~/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/sapolyakov535/git-extended/pull/new/develop
remote:
To github.com:sapolyakov535/git-extended.git
 * [new branch]      develop -> develop
sapolyakov@sapolyakov:~/git-extended$
```

Рисунок 3.6: Инициализация и загрузка изменений

Установите внешнюю ветку как вышестоящую для этой ветки

Создадим релиз с версией 1.0.0

Создадим журнал изменений

Добавим журнал изменений в индекс

Зальём релизную ветку в основную ветку

```
branch 'develop' set up to track 'origin/develop'.
sapolyakov@sapolyakov:~/git-extended$ git flow release start 1.0.0
Switched to a new branch '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'

sapolyakov@sapolyakov:~/git-extended$ standard-changelog --first-release
sapolyakov@sapolyakov:~/git-extended$ git add CHANGELOG.md
sapolyakov@sapolyakov:~/git-extended$ git commit -am 'chore(site): add changelog'
[release/1.0.0 76ee748] chore(site): add changelog
1 file changed, 4 insertions(+)
create mode 100644 CHANGELOG.md
sapolyakov@sapolyakov:~/git-extended$ git flow release finish 1.0.0
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
Merge made by the 'ort' strategy.
CHANGELOG.md | 4 +++
1 file changed, 4 insertions(+)
create mode 100644 CHANGELOG.md
Already on 'master'
Your branch is ahead of 'origin/master' by 2 commits.
(use "git push" to publish your local commits)
Switched to branch 'develop'
```

Рисунок 3.7: Завершение релиза

Отправим данные на github

```
sapolyakov@sapolyakov:~/git-extended$  
sapolyakov@sapolyakov:~/git-extended$ git push --all  
Enumerating objects: 6, done.  
Counting objects: 100% (6/6), done.  
Delta compression using up to 8 threads  
Compressing objects: 100% (4/4), done.  
Writing objects: 100% (5/5), 2.67 KiB | 2.67 MiB/s, done.  
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)  
To github.com:sapolyakov535/git-extended.git  
    f8f7241..6fb3194  develop -> develop  
    f8f7241..626745e  master -> master  
sapolyakov@sapolyakov:~/git-extended$ git push --tags  
Enumerating objects: 1, done.  
Counting objects: 100% (1/1), done.  
Writing objects: 100% (1/1), 158 bytes | 158.00 KiB/s, done.  
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)  
To github.com:sapolyakov535/git-extended.git  
* [new tag]      v1.0.0 -> v1.0.0  
sapolyakov@sapolyakov:~/git-extended$ gh release create v1.0.0 -F CHANGELOG.md  
https://github.com/sapolyakov535/git-extended/releases/tag/v1.0.0  
sapolyakov@sapolyakov:~/git-extended$
```

Рисунок 3.8: Отправка

Создадим ветку для новой функциональности По окончании разработки новой функциональности следующим шагом следует объединить ветку feature\_branch с develop:

```
sapolyakov@sapolyakov:~/git-extended$ git flow feature start feature_branch
Switched to a new branch '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

sapolyakov@sapolyakov:~/git-extended$ touch test2
sapolyakov@sapolyakov:~/git-extended$ git flow feature finish feature_branch
Switched to branch 'develop'
Your branch is up to date with 'origin/develop'.
Already up to date.
Deleted branch feature/feature_branch (was 6fb3194).

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'
```

Рисунок 3.9: Объединение веток

Создадим релиз с версией 1.2.3

Обновите номер версии в файле package.json. Установите её в 1.2.3

Создадим журнал изменений

Добавим журнал изменений в индекс

Зальём релизную ветку в основную ветку

```
Summary of actions:
- Release branch 'release/1.2.3' has been merged into 'master'
- The release was tagged 'v1.2.3'
- Release tag 'v1.2.3' has been back-merged into 'develop'
- Release branch 'release/1.2.3' has been locally deleted
- You are now on branch 'develop'

sopolyakov@sopolyakov:~/git-extended$ git push --all
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 2.79 KiB | 2.79 MiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:sopolyakov535/git-extended.git
  6fb3194..add8eda  develop -> develop
  626745e..5ded4d7  master -> master
sopolyakov@sopolyakov:~/git-extended$ git push --tags
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 158 bytes | 158.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:sopolyakov535/git-extended.git
 * [new tag]      v1.2.3 -> v1.2.3
sopolyakov@sopolyakov:~/git-extended$ gh release create v1.2.3 -F CHANGELOG.md
https://github.com/sopolyakov535/git-extended/releases/tag/v1.2.3
sopolyakov@sopolyakov:~/git-extended$
```

Рисунок 3.10: Завершение релиза

## 3.2 Подготовка рабочего репозитория

```
sapolyakov@sapolyakov:/work/study/2025-2026/Операционные системы/2026-1--study--os-intro$ git add .
sapolyakov@sapolyakov:/work/study/2025-2026/Операционные системы/2026-1--study--os-intro$ git cz
cz-cli@4.3.1, cz-conventional-changelog@3.3.0

? Select the type of change that you're committing: chore: Other changes that don't modify src or test files
? What is the scope of this change (e.g. component or file name): (press enter to skip)
? Write a short, imperative tense description of the change (max 93 chars):
(4) test
? 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
[master 0a0aa9a] chore: test
 1 file changed, 11 insertions(+), 11 deletions(-)
sapolyakov@sapolyakov:/work/study/2025-2026/Операционные системы/2026-1--study--os-intro$ git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 1.12 KiB | 1.12 MiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:sapolyakov535/2026-1--study--os-intro.git
 d8b06df..0a0aa9a master -> master
sapolyakov@sapolyakov:/work/study/2025-2026/Операционные системы/2026-1--study--os-intro$
```

Рисунок 3.11: package.json и коммит

```
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'

sapolyakov@sapolyakov:~/work/study/2025-2026/Операционные системы/2026-1--study--os-intro$ git push --all
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 2.75 KiB | 2.75 MiB/s, done.
Total 5 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:sapolyakov535/2026-1--study--os-intro.git
  0a0aa9a..fe77228 develop -> develop
  0a0aa9a..6e71835 master -> master
sapolyakov@sapolyakov:~/work/study/2025-2026/Операционные системы/2026-1--study--os-intro$ git push --tags
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 156 bytes | 156.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:sapolyakov535/2026-1--study--os-intro.git
 * [new tag]      v1.0.0 -> v1.0.0
sapolyakov@sapolyakov:~/work/study/2025-2026/Операционные системы/2026-1--study--os-intro$
```

Рисунок 3.12: Завершение релиза

## **4 Вывод**

Мы приобрели практические навыки взаимодействия с дополнительными функциями гитхаб.