

# **Отчет по лабораторной работе №4**

**Операционные системы**

Абронина Алиса Кирилловна

# 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>
<b>4</b>	<b>Установка ПО. Установка git-glow, node.js. Настройка node.js.. Общепринятые коммиты.</b>	<b>8</b>
4.1	Практический сценарий использования git. Создание репозитория. Работа с репозиторием. . . . .	10
<b>5</b>	<b>Выводы</b>	<b>19</b>

# Список иллюстраций

4.1	Установка git-glow . . . . .	8
4.2	Установка node.js . . . . .	8
4.3	Настройка node.js . . . . .	9
4.4	Настройка commitizen, standard-changelog . . . . .	9
4.5	Настройка commitizen, standard-changelog . . . . .	10
4.6	репозиторий . . . . .	11
4.7	Конфигурация коммитов . . . . .	11
4.8	Отправка изменений на github . . . . .	12
4.9	ч1 . . . . .	13
4.10	ч2 . . . . .	13
4.11	ч3 . . . . .	14
4.12	ч1 . . . . .	14
4.13	ч2 . . . . .	15
4.14	ч3 . . . . .	15
4.15	Создание и слияние веток . . . . .	16
4.16	Изменение файла . . . . .	16
4.17	Ч1 . . . . .	17
4.18	Ч2 . . . . .	17
4.19	Ч3 . . . . .	18

## Список таблиц

# 1 Цель работы

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 Выполнение лабораторной работы**

## 4 Установка ПО. Установка git-glow, node.js. Настройка node.js..

### Общепринятые коммиты.

Устанавливаем git-glow (рис. 4.1).

```
[akabronina@vbox tmp]$ wget --no-cleck-certificate -q https://raw.githubusercontent.com/petervanderdoes/gitflow/develop/contrib/gitflow-installer.sh
Unknown option 'cleck-certificate'
[akabronina@vbox tmp]$ wget --no-check-certificate -q https://raw.githubusercontent.com/petervanderdoes/gitflow/develop/contrib/gitflow-installer.sh
[akabronina@vbox tmp]$ chmod +x gitflow-installer.sh install stable
chmod: невозможно получить доступ к 'install': Нет такого файла или каталога
chmod: невозможно получить доступ к 'stable': Нет такого файла или каталога
[akabronina@vbox tmp]$ chmod +x gitflow-installer.sh
[akabronina@vbox tmp]$ sudo ./gitflow-installer.sh install stable
[sudo] пароль для akabronina: 
```

Рис. 4.1: Установка git-glow

Устанавливаем node.js (рис. 4.2).

```
[akabronina@vbox tmp]$ sudo dnf install nodejs
Обновление и загрузка репозитория:
```

Рис. 4.2: Установка node.js

Для работы с node.js добавим каталог с исполняемыми файлами, устанавливаем yarn, в переменную path (рис. 4.3).



```
[akabronina@vbox tmp]$ sudo dnf install pnpm
Обновление и загрузка репозитория:
```

Рис. 4.3: Настройка node.js

Настраиваем commitizen, standard-changelog (рис. 4.5). (рис. 4.4).

```
[akabronina@vbox tmp]$ pnpm add -g commitizen

Update available! 9.13.0 -> 10.5.2.
Changelog: https://github.com/pnpm/pnpm/releases/tag/v10.5.2
Run "pnpm self-update" to update.

Follow @pnpmjs for updates: https://x.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

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

Done in 1m 10.1s
[akabronina@vbox tmp]$ pnpm add -g standard-changelog
Progress: resolved 1, reused 1, downloaded 0, added 0
```

Рис. 4.4: Настройка commitizen, standard-changelog

```
*):$PNPM_HOME:*)" ;;  
*) export PATH="$PNPM_HOME:$PATH" ;;  
esac  
  
To start using pnpm, run:  
source /home/akabronina/.bashrc  
[akabronina@vbox tmp]$ source ~/bashrc  
bash: source: команда не найдена  
[akabronina@vbox tmp]$ source ~/bashrc  
bash: /home/akabronina/bashrc: Нет такого файла или каталога  
[akabronina@vbox tmp]$ source ~/.bashrc  
[akabronina@vbox tmp]$ pnpm add -g commitizen  
  
Update available! 9.13.0 → 10.5.2.  
Changelog: https://github.com/pnpm/pnpm/releases/tag/v10.5.2  
Run "pnpm self-update" to update.  
  
Follow @pnpmjs for updates: https://x.com/pnpmjs  
  
Progress: resolved 47, reused 0, downloaded 11, added 0
```

Рис. 4.5: Настройка commitizen, standard-changelog

## 4.1 Практический сценарий использования git.

### Создание репозитория. Работа с репозиторием.

Создаем репозиторий на Github. Делаем первый коммит и выкладываем на github (рис. 4.6).

```

[akabronina@vbox git-extended]$ git push -u origin master
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:   https://github.com/akabronina/git-extended/pull/new/master
remote:
To https://github.com/akabronina/git-extended.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
[akabronina@vbox git-extended]$ ls
README.md
[akabronina@vbox git-extended]$ git push -u
branch 'master' set up to track 'origin/master'.
[akabronina@vbox git-extended]$

```

Рис. 4.6: репозиторий

Конфигурация общепринятых коммитов. Для этого добавим в файл package.json команду для формирования коммитов (рис. 4.7).

```

package.json  [-M--] 0 L: [ 1+1/ 18/ 19] *(441 / 443b) 0125 0x
{
  "name": "git-extended",
  "version": "1.0.0",
  "description": "Git repo for educational purposes",
  "main": "index.js",
  "repository": "git@github.com:akabronina/git-extended.git",
  "scripts": {
    "test": "echo '\\Error: no test specified\\' && exit 1"
  },
  "keywords": [],
  "author": "Abronina Alisa alisaabronina@gmail.com",
  "license": "CC-BY-4.0",
  "config": {
    "commitizen": {
      "path": "cz-conventional-changelog"
    }
  }
}

```

Рис. 4.7: Конфигурация коммитов

Добавляем файлы, выполняем коммит, отправляем на github (рис. 4.8).

```

foot
[akabronina@vbox git-extended]$ git add .
[akabronina@vbox git-extended]$ git cz
Parsing JSON at /home/akabronina/git-extended/package.json for co
mmitizen config failed:

[akabronina@vbox git-extended]$ mc

[akabronina@vbox git-extended]$ prnm install -g commitizen
bash: prnm: команда не найдена
[akabronina@vbox git-extended]$ npm install -g commitizen
.:^C
[akabronina@vbox git-extended]$ git commit -m "fix: "
[master 1cb7cf0] fix:
 1 file changed, 18 insertions(+)
 create mode 100644 package.json
[akabronina@vbox git-extended]$ git push
Перечисление объектов: 4, готово.
Подсчет объектов: 100% (4/4), готово.
Сжатие объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 1.19 КиБ | 1.19 МиБ/с, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/akabronina/git-extended.git
   549a646..1cb7cf0  master -> master
[akabronina@vbox git-extended]$ [akabronina@vbox git-extended][ak
[akabronina@vbox git-extended]$ git flow

```

Рис. 4.8: Отправка изменений на github

Инициализируем git-flow. Префикс для ярлыков установим v. Проверяем, что мы работает на ветке develop. Загружаем весь репозиторий, устанавливаем внешнюю ветку как вышестоящую для этой ветки. Создадим с версией 1.0.0 (рис. 4.9). (рис. 4.10). (рис. 4.11).

```

foot
[akabronina@vbox git-extended]$ git add .
[akabronina@vbox git-extended]$ git cz
Parsing JSON at /home/akabronina/git-extended/package.json for co
mmitizen config failed:

[akabronina@vbox git-extended]$ mc

[akabronina@vbox git-extended]$ pnm install -g commitizen
bash: pnm: команда не найдена
[akabronina@vbox git-extended]$ npm install -g commitizen
.:^C
[akabronina@vbox git-extended]$ git commit -m "fix: "
[master 1cb7cf0] fix:
 1 file changed, 18 insertions(+)
 create mode 100644 package.json
[akabronina@vbox git-extended]$ git push
Перечисление объектов: 4, готово.
Подсчет объектов: 100% (4/4), готово.
Сжатие объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 1.19 КиБ | 1.19 МиБ/с, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/akabronina/git-extended.git
   549a646..1cb7cf0  master -> master
[akabronina@vbox git-extended]$ [akabronina@vbox git-extended][ak
[akabronina@vbox git-extended]$ git flow

```

Рис. 4.9: ч1

```

[akabronina@vbox git-extended]$ git flow init

Which branch should be used for bringing forth production release
s?
  - 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/akabronina/git-extended/.git/
hooks]
[akabronina@vbox git-extended]$ git branch
* develop
  master

```

Рис. 4.10: ч2

```
[akabronina@vbox 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/akabronina/git-extended/pull/new/develop
remote:
To https://github.com/akabronina/git-extended.git
 * [new branch]      develop -> develop
```

Рис. 4.11: ч3

Создадим журнал изменений. Добавим журнал изменений в индекс. Затем релизную ветку ветку в основную ветку. Отправим данные на github. Создадим релиз на github. Для этого будем использовать утилиты с github (рис. 4.12).(рис. 4.13).(рис. 4.14).

```
[akabronina@vbox 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'

[akabronina@vbox git-extended]$ tsandarf-changelog --first-release
bash: tsandarf-changelog: команда не найдена
[akabronina@vbox git-extended]$ standard-changelog --first-release
✓ created CHANGELOG.md
✓ output changes to CHANGELOG.md
[akabronina@vbox git-extended]$ fit add CHANGELOG.md
bash: fit: команда не найдена
[akabronina@vbox git-extended]$ git add CHANGELOG.md
[akabronina@vbox git-extended]$ git comm
```

Рис. 4.12: ч1



```

Fatal: Tagging failed. Please run finish again to retry.
[akabronina@vbox git-extended]$ git flow release finish 1.0.0
Branches 'master' and 'origin/master' have diverged.
And local branch 'master' is ahead of 'origin/master'.
Уже на «master»
Ваша ветка опережает «origin/master» 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 | 4 ++++
  1 file changed, 4 insertions(+)
  create mode 100644 CHANGELOG.md
Ветка release/1.0.0 удалена (была 145f21d).

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'

[akabronina@vbox git-extended]$ git push --all

```

Рис. 4.13: ч2

```

[akabronina@vbox git-extended]$ git push --all
Перечисление объектов: 6, готово.
Подсчет объектов: 100% (6/6), готово.
Сжатие объектов: 100% (4/4), готово.
Запись объектов: 100% (5/5), 2.67 КиБ | 2.67 МБ/с, готово.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/akabronina/git-extended.git
   1cb7cf0..5092893  develop -> develop
   1cb7cf0..0dbbd92  master -> master
[akabronina@vbox git-extended]$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 166 байтов | 166.00 КиБ/с, готово.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/akabronina/git-extended.git
   * [new tag]           V1.0.0 -> V1.0.0
[akabronina@vbox git-extended]$ gh release create v1.0.0 -F CHANGELOG.md
https://github.com/akabronina/git-extended/releases/tag/v1.0.0

```

Рис. 4.14: ч3

Создадим ветку для новой функциональности. Следующим шагом следует объединить ветки feature-branch с develop(рис. 4.15).

```

[akabronina@vbox git-extended]$ git flow feature finish feature_b
ranch
Переключились на ветку «develop»
Эта ветка соответствует «origin/develop».
Уже актуально.
Ветка feature/feature_branch удалена (была 5092893).

Summary of actions:
- The feature branch 'feature/feature_branch' was merged into 'de
velop'
- Feature branch 'feature/feature_branch' has been locally delete
d
- You are now on branch 'develop'
[akabronina@vbox git-extended]$ git flow re

```

Рис. 4.15: Создание и слияние веток

Создадим релиз с версией 1.2.3. Обновим номер версии в файле package.json и установим ее в 1.2.3 (рис. 4.16).

```

mc [akabronina@vbox]:~/git-extended
package.json  [----] 19 L: [ 1+ 2 3/ 19] *(47 / 443[*])[X
(
  "name": "git-extended",
  "version": "1.2.3",
  "description": "Git repo for educational purposes",
  "main": "index.js",
  "repository": "git@github.com:akabronina/git-extended.git",
  "scripts": {
    "test": "echo '\\Error: no test specified\\' && exit 1"
  },
  "keywords": [],
  "author": "Abronina Alisa alisasbronina@gmail.com",
  "license": "CC-BY-4.0",
  "config": {
    "commitizen": {
      "path": "cz-conventional-changelog"
    }
  }
)

```

Рис. 4.16: Изменение файла

Создадим журнал изменений, добавим журнал изменений в индекс, зальем релизную ветку в основную ветку, отправляем данные на github, создадим релиз с комментарием из журнала изменений (рис. 4.17).(рис. 4.18). (рис. 4.19).



```

root
hing model.
  feature  Manage your feature branches.
  bugfix   Manage your bugfix branches.
  release  Manage your release branches.
  hotfix   Manage your hotfix branches.
  support  Manage your support branches.
  version  Shows version information.
  config   Manage your git-flow configuration.
  log      Show log deviating from base branch.

Try 'git flow <subcommand> help' for details.
[akabronina@vbox 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: Ч1

```

root
Переключились на ветку «master»
Эта ветка соответствует «origin/master».
Merge made by the 'ort' strategy.
 CHANGELOG.md | 4 ++++
 package.json | 2 +-
 2 files changed, 5 insertions(+), 1 deletion(-)
Уже на «master»
Ваша ветка опережает «origin/master» на 3 коммита.
(используйте «git push», чтобы опубликовать ваши локальные комм
иты)
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 30f6a6d).

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'

```

Рис. 4.18: Ч2

```

foot
Ветка release/1.2.3 удалена (была 30f6a6d).

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'

[akabronina@vbox git-extended]$ git push --all
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
Сжатие объектов: 100% (5/5), готово.
Запись объектов: 100% (6/6), 2.72 КиБ | 2.72 МБ/с, готово.
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 https://github.com/akabronina/git-extended.git
   5092893..d05e1ed  develop -> develop
   0dbbd92..fd57772  master -> master
[akabronina@vbox git-extended]$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 169 байтов | 169.00 КиБ/с, готово.

```

Рис. 4.19: ЧЗ

## 5 Выводы

Я получила навыки правильной работы с репозиториями git