

# Лабораторная работа 4

## Продвинутое использование git

Богданюк Анна Васильевна

### Содержание

1	Цель работы .....	1
2	Задание.....	1
3	Теоретическое введение .....	1
4	Выполнение лабораторной работы .....	2
5	Выводы.....	10

### 1 Цель работы

Получение навыков правильной работы с репозиториями git.

### 2 Задание

1. Установка программного обеспечения
2. Установка git-flow
3. Установка Node.js
4. Настройка Node.js
5. Общепринятые коммиты

### 3 Теоретическое введение

Gitflow Workflow опубликована и популяризована Винсентом Дриссенем. 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 сливается с веткой develop. Когда работа над веткой релиза release завершена, она сливается в ветки develop и master. Если в master обнаружена проблема, из master создаётся ветка hotfix. Когда работа над веткой исправления hotfix завершена, она сливается в ветки develop и master.

## 4 Выполнение лабораторной работы

Для того, чтобы установить необходимое программное обеспечение, сначала скачаю обновлённые версии пакета (рис. 1).

```
avbogdanyuk@Bogdanyuk:~$ sudo apt-get update
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Hit:2 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:3 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1239 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [222 kB]
Get:6 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1239 kB]
```

### Обновления

Затем уже устанавливаю git-flow (рис. 2).

```
avbogdanyuk@Bogdanyuk:~$ sudo apt-get install git-flow
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

### Установка git-flow

Теперь необходимо установить Node.js. На Node.js базируется программное обеспечение для семантического версионирования и общепринятых коммитов. (рис. 3).

Using Ubuntu	<pre>curl -fsSL https://deb.nodesource.com/setup_21.x &gt;   sudo apt-get install -y nodejs</pre>
Using Debian, as root	<pre>curl -fsSL https://deb.nodesource.com/setup_21.x &gt;   apt-get install -y nodejs</pre>
Node.js v20.x:	
Using Ubuntu	<pre>curl -fsSL https://deb.nodesource.com/setup_20.x &gt;   sudo apt-get install -y nodejs</pre>
Using Debian, as root	

```
Hit:4 http://archive.ubuntu.com/ubuntu jammy-backports InRelease
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ca-certificates is already the newest version (20230311ubuntu0.22.04.1).
ca-certificates set to manually installed.
curl is already the newest version (7.81.0-1ubuntu1.15).
curl set to manually installed.
gnupg is already the newest version (2.2.27-3ubuntu2.1).
gnupg set to manually installed.
The following NEW packages will be installed:
  apt-transports-https
0 upgraded, 1 newly installed, 0 to remove and 62 not upgraded.
Need to get 1510 B of archives.
After this operation, 170 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 apt-transports-https
  1510 B in 0s (5180 B/s)
Selecting previously unselected package apt-transports-https.
(Reading database ... 320108 files and directories currently installed.)
Preparing to unpack .../apt-transports-https_2.4.11_all.deb ...
Unpacking apt-transports-https (2.4.11) ...
Setting up apt-transports-https (2.4.11) ...
gpg: WARNING: unsafe ownership on homedir '/home/avbogdanyuk/.gnupg'
Hit:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:2 https://deb.nodesource.com/node_21.x nodistro InRelease [12.1 kB]
Hit:3 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:4 https://deb.nodesource.com/node_21.x nodistro/main amd64 Packages [3002 B]
Hit:5 http://archive.ubuntu.com/ubuntu jammy-updates InRelease
```

### Установка Node.js

Затем подгружаю rprmt (рис. 4).

```

avbogdanyuk@Bogdanyuk:~$ sudo apt-get install nodejs npm
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nodejs is already the newest version (21.7.0-1nodesource1).
Some packages could not be installed. This may mean that you have
requested an impossible situation or if you are using the unstable
distribution that some required packages have not yet been created
or been moved out of Incoming.
The following information may help to resolve the situation:

The following packages have unmet dependencies:
nodejs : Conflicts: npm
npm : Depends: node-cacache but it is not going to be installed
       Depends: node-gyp but it is not going to be installed
       Depends: node-mkdirp but it is not going to be installed
       Depends: node-nopt but it is not going to be installed
       Depends: node-tar but it is not going to be installed
       Depends: node-which but it is not going to be installed
       Depends: nodejs:any (>= 10)
       Recommends: node-tap but it is not going to be installed
E: Unable to correct problems, you have held broken packages.
avbogdanyuk@Bogdanyuk:~$ sudo npm install -g pnpm
(Progress bar) @ idealTree: timing idealTree Completed in 422ms

```

### Установка pnpm

Для работы с Node.js добавим каталог с исполняемыми файлами, устанавливаемыми yarn, в переменную PATH (рис. 5).

```

avbogdanyuk@Bogdanyuk:~$ pnpm setup
Appended new lines to /home/avbogdanyuk/.bashrc

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

To start using pnpm, run:
source /home/avbogdanyuk/.bashrc
avbogdanyuk@Bogdanyuk:~$

```

### Запускаю pnpm setup

Выполняю source ~/.bashrc (рис. 6).

```

To start using pnpm, run:
source /home/avbogdanyuk/.bashrc
avbogdanyuk@Bogdanyuk:~$ source /home/avbogdanyuk/.bashrc

```

### source ~/.bashrc

Данная программа используется для помощи в форматировании коммитов. При этом устанавливается скрипт git-cz, который мы и будем использовать для коммитов (рис. 7).

```

avbogdanyuk@Bogdanyuk:~$ pnpm add -g commitizen
Progress: resolved 14, reused 0, downloaded 13, added 0

```

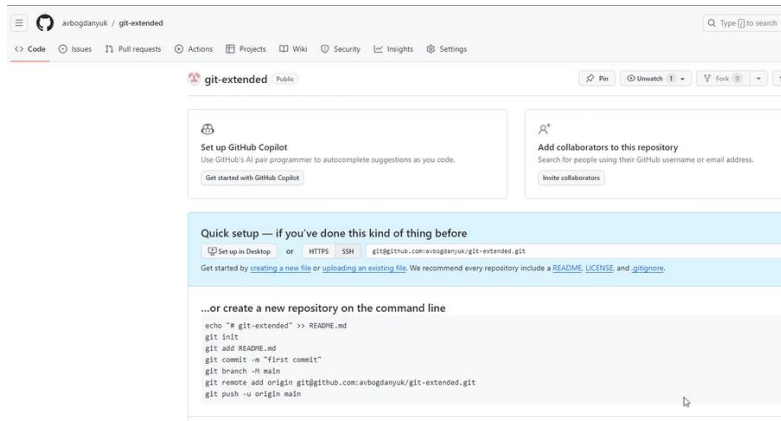
### Форматирование коммитов

Данная программа используется для помощи в создании логов (рис. 8).

```
avbogdanyuk@Bogdanyuk:~$ npm add -g standard-changelog
Progress: resolved 1, reused 1, downloaded 0, added 0
```

## Создание логов

Теперь переходим к практическому сценарию использования git. Для начала создаю репозиторий на GitHub git-extended (рис. 9).



## Созданный репозиторий git-extended

Делаем первый коммит и выкладываем на github (рис. 10).

```
echo "# git-extended" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin git@github.com:avbogdanyuk/git-extended.git
git push -u origin main
```

## Первый коммит

Конфигурация для пакетов Node.js (рис. 11).

```
avbogdanyuk@Bogdanyuk:~$ npm init
Wrote to /home/avbogdanyuk/package.json

{
  "name": "avbogdanyuk",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \\\"Error: no test specified\\\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}
avbogdanyuk@Bogdanyuk:~$
```

## Конфигурация для пакетов Node.js

Теперь необходимо заполнить несколько параметров пакета. Сконфигурируем формат коммитов. Для этого добавим в файл package.json команду для формирования коммитов (рис. 12).

```
GNU nano 6.2 /home/avbogdanyuk/package.json
{
  "name": "git-extended",
  "version": "1.0.0",
  "description": "Git repo for educational purposes",
  "main": "index.js",
  "repository": "git@github.com:avbogdanyuk/git-extended.git",
  "author": "Anna Bogdanyuk <silverblood260@gmail.com>",
  "license": "CC-BY-4.0",
  "config": {
    "commitizen": {
      "path": "cz-conventional-changelog"
    }
  }
}
```

## Конфигурация пакетов Node.js

Добавляем новый файл package.json (рис. 13).

```
avbogdanyuk@Bogdanyuk:~$ git add package.json
```

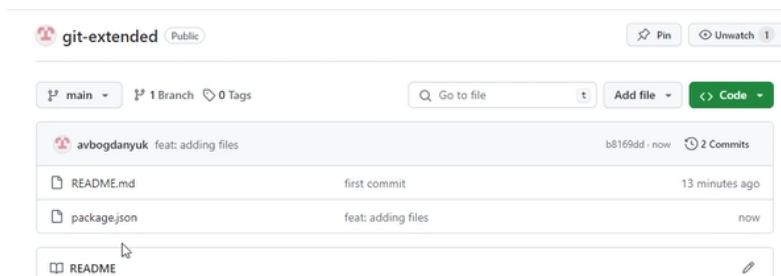
## Добавляем файл

Выполняем коммит. Я выбрала тип feat. И отправляем на github (рис. 14).

```
avbogdanyuk@Bogdanyuk:~$ 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)
? Write a short, imperative tense description of the change (max 94 chars):
(12) adding files
? Provide a longer description of the change: (press enter to skip)
? Test this on GitHub? (y/n)
```

## Выполняем коммит

Вид моего репозитория на гитхаб (рис. 15).



## Репозиторий

Инициализируем git-flow, префикс для ярлыков установим в v (рис. 16).

```

avbogdanyuk@bogdanyuk:~$ git flow init -f
Which branch should be used for bringing forth production releases?
  - develop
  - main
Branch name for production releases: [main]
Which branch should be used for integration of the "next release"?
  - develop
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/avbogdanyuk/.git/hooks]
avbogdanyuk@bogdanyuk:~$

```

*Инициализируем git-flow*

Проверяем, что я на ветке develop, все верно (рис. 17).

```

avbogdanyuk@bogdanyuk:~$ git branch
* develop
main

```

*Проверка ветки*

Загружаю весь репозиторий в хранилище (рис. 18).

```

avbogdanyuk@bogdanyuk:~$ git push --all
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:   https://github.com/avbogdanyuk/git-extended/pull/new/develop
remote:
To github.com:avbogdanyuk/git-extended.git
 * [new branch]   develop -> develop
avbogdanyuk@bogdanyuk:~$

```

*Загрузка всего репозиторий*

Устанавливаем внешнюю ветку как вышестоящую для этой ветки (рис. 19).

```

avbogdanyuk@bogdanyuk:~$ git branch --set-upstream-to=origin/develop develop
Branch 'develop' set up to track remote branch 'develop' from 'origin'.

```

*Устанавливаем иерархию*

Создаем релиз с версией 1.0.0 (рис. 20).

```

avbogdanyuk@bogdanyuk:~$ 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'

```

*Создаем первый релиз*



Создаем журнал изменений (рис. 21).

```
avbogdanyuk@bogdanyuk:~$ standard-changelog --first-release
3 created CHANGELOG.md
3 output changes to CHANGELOG.md
```

*Журнал изменений*

Добавим журнал изменений в индекс (рис. 22).

```
avbogdanyuk@bogdanyuk:~$ git add CHANGELOG.md
avbogdanyuk@bogdanyuk:~$ git commit -am 'chore(site): add changelog'
[release/1.0.0 63f2e79] chore(site): add changelog
1 file changed, 9 insertions(+)
create mode 100644 CHANGELOG.md
```

*Добавляем журнал изменений*

Зальём релизную ветку в основную ветку (рис. 23).

```
avbogdanyuk@bogdanyuk:~$ git flow release finish 1.0.0_
```

*Из релизной ветки в основную*

Отправляю данные на github (рис. 24).

```
avbogdanyuk@bogdanyuk:~$ git push --all
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 12 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 708 bytes | 708.00 KiB/s, done.
Total 5 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), done.
To github.com:avbogdanyuk/git-extended.git
   b8169dd..367e8ac develop -> develop
   b8169dd..b9cfaac main -> main
avbogdanyuk@bogdanyuk:~$ git push --tags
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 171 bytes | 171.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:avbogdanyuk/git-extended.git
 * [new tag]         v1.0.0 -> v1.0.0
```

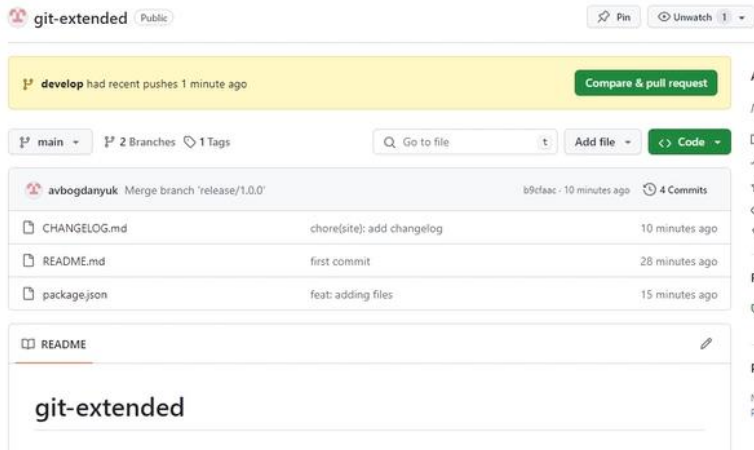
*Отправка на github*

Создаю релиз на гитхаб. Для этого использую утилит работы с гитхаб (рис. 25).

```
avbogdanyuk@bogdanyuk:~$ gh release create v1.0.0 -F CHANGELOG.md
2024/03/08 22:06:22.949344 cmd_run.go:1055: WARNING: cannot start document portal: dial unix /run/user/1000/bus: connect
: no such file or directory
https://github.com/avbogdanyuk/git-extended/releases/tag/v1.0.0
```

*Создание релиза*

Вид моего гитхаба на этот момент выполнения лабораторной. Можео заметить, что появился tag и 2 ветки (рис. 26).



## Репозиторий

Теперь перейдем к работе с репозиторием git. Начнем с разработки новой функциональности. Создаю ветку для новой функциональности (рис. 27).

```
avbogdanyuk@Bogdanyuk:~$ 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

avbogdanyuk@Bogdanyuk:~$
```

## Новая ветка

Далее, продолжаю работу с гит как обычно. По окончании разработки новой функциональности следующим шагом следует объединить ветку feature\_branch с develop (рис. 28).

```
avbogdanyuk@Bogdanyuk:~$ 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 367e8ac).

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'

avbogdanyuk@Bogdanyuk:~$
```

## Объединяю feature\_branch с develop

Следующий шаг - создание релиза git-flow. Создадим для начала релиз с версией 1.2.3 (рис. 29).



```

avbogdanyuk@Bogdanyuk:~$ git flow release start 1.2.3
Switched to a new branch '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'

```

## Новый релиз

Обновляю номер версии в файле package.json. Устанавливаю её в 1.2.3 (рис. 30).

```

GNU nano 6.2 /home/avbogdanyuk/package.json
{
  "name": "git-extended",
  "version": "1.2.3"
}

```

Обновляю номер версии в файле package.json

Создаю журнал изменений (рис. 31).

```

avbogdanyuk@Bogdanyuk:~$ standard-changelog
output changes to CHANGELOG.md

```

## Журнал изменений

Добавляю журнал изменений в индекс (рис. 32).

```

avbogdanyuk@Bogdanyuk:~$ git add CHANGELOG.md
avbogdanyuk@Bogdanyuk:~$ git commit -am 'chore(site): update changelog'
[release/1.2.3 f45f49b] chore(site): update changelog
2 files changed, 5 insertions(+), 1 deletion(-)

```

Добавление журнала изменений в индекс

Заливаю релизную ветку в основную (рис. 33).

```

avbogdanyuk@Bogdanyuk:~$ git flow release finish 1.2.3
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
Merge made by the 'ort' strategy.
  CHANGELOG.md | 4 +++
  package.json | 2 +-
  2 files changed, 5 insertions(+), 1 deletion(-)
Already on 'main'
Your branch is ahead of 'origin/main' by 3 commits.
(use "git push" to publish your local commits)
Switched to branch 'develop'
Your branch is up to date with 'origin/develop'.
Merge made by the 'ort' strategy.
  CHANGELOG.md | 4 +++
  package.json | 2 +-
  2 files changed, 5 insertions(+), 1 deletion(-)
Deleted branch release/1.2.3 (was f45f49b).

Summary of actions:
- Release branch 'release/1.2.3' has been merged into 'main'
- 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'

avbogdanyuk@Bogdanyuk:~$

```

Из релизной в основную

Отправляю данных на github (рис. 34).

```
avbogdanyuk@bogdanyuk:~$ git push --all
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 12 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 784 bytes | 784.00 KiB/s, done.
Total 6 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (3/3), completed with 2 local objects
To github.com:avbogdanyuk/git-extended.git
   367e8ac..349c1c1  develop -> develop
   b9cfaac..32ae15f  main -> main
avbogdanyuk@bogdanyuk:~$ git push --tags
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 180 bytes | 180.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:avbogdanyuk/git-extended.git
 * [new tag]         v1.2.3 -> v1.2.3
avbogdanyuk@bogdanyuk:~$
```

Отправка на github

Создаю релиз на github с комментарием из журнала изменений (рис. 35).

```
avbogdanyuk@bogdanyuk:~$ gh release create v1.2.3 -F CHANGELOG.md
2024/03/08 22:13:03.303197 cmd_run.go:1055: WARNING: cannot start document portal: dial unix /run/user/1000/bus: connect: no such file or direct
https://github.com/avbogdanyuk/git-extended/releases/tag/v1.2.3
avbogdanyuk@bogdanyuk:~$
```

Создание нового релиза с комментарием из журнала изменений

## 5 Выводы

В ходе выполнения лабораторной работы были получены навыки правильной работы с репозиториями git.