

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 4 _____

дисциплина: *Операционные системы*

Студент: Ниemek Яи Жак

Группа: НММБд-04-24

МОСКВА

2025__ г.

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.

Последовательность выполнения работы

Установка программного обеспечения

Установка git-flow

- Linux
 - Fedora
 - Установка из коллекции репозитория Copr (<https://copr.fedorainfracloud.org/coprs/elegos/gitflow/>):
 - # Enable the copr repository
 - dnf copr enable elegos/gitflow
 - # Install gitflow
 - dnf install gitflow

Установка Node.js

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

- Fedora
 - dnf install nodejs
 - dnf install pnpm

Настройка Node.js

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

- Запустите:
 - pnpm setup
- Перелогиньтесь, или выполните:
 - source ~/.bashrc

Общепринятые коммиты

1. commitizen

- Данная программа используется для помощи в форматировании коммитов.
- pnpm add -g commitizen
- При этом устанавливается скрипт git-cz, который мы и будем использовать для коммитов.

2. standard-changelog

- Данная программа используется для помощи в создании логов.
- pnpm add -g standard-changelog

Практический сценарий использования git

Создание репозитория git

1. Подключение репозитория к github

- Создайте репозиторий на GitHub. Для примера назовём его git-extended.
- Делаем первый коммит и выкладываем на github:
- `git commit -m "first commit"`
- `git remote add origin git@github.com:<username>/git-extended.git`
- `git push -u origin master`

2. Конфигурация общепринятых коммитов

- Конфигурация для пакетов Node.js
- `pnpm init`

Необходимо заполнить несколько параметров пакета.

- Название пакета.
- Лицензия пакета. Список лицензий для npm: <https://spdx.org/licenses/>. Предлагается выбрать лицензию CC-BY-4.0.
- Сконфигурируем формат коммитов. Для этого добавим в файл `package.json` команду для формирования коммитов:
- `"config": {`
- `"commitizen": {`
- `"path": "cz-conventional-changelog"`
- `}`
- `}`

Таким образом, файл `package.json` приобретает вид:

```
{
  "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": "Name Surname <username@gmail.com>",
  "license": "CC-BY-4.0",
  "config": {
    "commitizen": {
      "path": "cz-conventional-changelog"
    }
  }
}
```

- Добавим новые файлы:
- `git add .`
- Выполним коммит:
- `git cz`
- Отправим на github:
- `git push`

3. Конфигурация git-flow

- Инициализируем git-flow
- `git flow init`

Префикс для ярлыков установим в `v`.

- Проверьте, что Вы на ветке `develop`:
- `git branch`
- Загрузите весь репозиторий в хранилище:

- `git push --all`
- 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'`
- Зальём релизную ветку в основную ветку
- `git flow release finish 1.0.0`
- Отправим данные на github
- `git push --all`
- `git push --tags`
- Создадим релиз на github. Для этого будем использовать утилиты работы с github:
- `gh release create v1.0.0 -F CHANGELOG.md`

Работа с репозиторием git

1. Разработка новой функциональности

- Создадим ветку для новой функциональности:
- `git flow feature start feature_branch`
- Далее, продолжаем работу с git как обычно.
- По окончании разработки новой функциональности следующим шагом следует объединить ветку `feature_branch` с `develop`:
- `git flow feature finish feature_branch`

2. Создание релиза git-flow

- 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'`
- Зальём релизную ветку в основную ветку
- `git flow release finish 1.2.3`
- Отправим данные на github
- `git push --all`
- `git push --tags`
- Создадим релиз на github с комментарием из журнала изменений:

`gh release create v1.2.3 -F CHANGELOG.md`

```

jacques@vbox:~$ sudo dnf copr enable elegos/gitflow
[sudo] Mot de passe de jacques :
Activation d'un dépôt Copr. Veuillez noter que ce dépôt n'est pas
partie intégrante de la distribution, et que la qualité pourrait varier.

Le projet Fedora n'exerce aucun pouvoir sur le contenu de ce dépôt au delà
des règles précisées dans la FAQ Copr
<https://docs.pagure.org/copr.copr/user_documentation.html#what-i-can-build-in-copr>,
et les paquets ne sont tenus à aucun niveau de qualité ou de sécurité.

Veuillez ne pas signaler de bogues à propos de ces paquets dans le Bugzilla de Fedora.
En cas de problèmes, contactez le propriétaire de ce dépôt.

Do you really want to enable copr.fedorainfracloud.org/elegos/gitflow? [y/N]: y
Erreur : It wasn't possible to enable this project.
Le dépôt « fedora-39-x86_64 » n'existe pas dans le projet « elegos/gitflow ».
Dépôts disponibles : 'fedora-42-aarch64', 'fedora-40-aarch64', 'fedora-rawhide-aarch64', 'fedora-42-x86_64', 'fedora-41-a
arch64', 'fedora-40-x86_64', 'fedora-43-x86_64', 'fedora-43-aarch64', 'fedora-41-x86_64', 'fedora-rawhide-x86_64'

If you want to enable a non-default repository, use the following command:
'dnf copr enable elegos/gitflow <repository>'
But note that the installed repo file will likely need a manual modification.
jacques@vbox:~$

```

```

jacques@vbox:~$ sudo dnf install gitflow
Fedora 39 - x86_64 - Updates                               6.2 kB/s | 5.0 kB    00:00
Dernière vérification de l'expiration des métadonnées effectuée il y a 0:00:05 le mer. 27 août 2025 18:20:03.
Aucune correspondance pour le paramètre: gitflow
Erreur : Impossible de trouver une correspondance: gitflow
jacques@vbox:~$ sudo dnf install nodejs
Dernière vérification de l'expiration des métadonnées effectuée il y a 0:00:43 le jeu. 28 août 2025 05:04:11.
Le paquet nodejs-1:20.17.0-1.fc39.x86_64 est déjà installé.
Dépendances résolues.
Rien à faire.
Terminé !
jacques@vbox:~$ sudo dnf install pnpm
Dernière vérification de l'expiration des métadonnées effectuée il y a 0:00:59 le jeu. 28 août 2025 05:04:11.
Dépendances résolues.
=====
Paquet                Architecture      Version           Dépôt             Taille
=====
Installation:
pnpm                  noarch            8.12.0-1.fc39     updates           2.6 M

Résumé de la transaction
=====
Installer 1 Paquet

Taille totale des téléchargements : 2.6 M
Taille des paquets installés : 13 M
Voulez-vous continuer ? [o/N] : o
Téléchargement des paquets :
[MIRROR] pnpm-8.12.0-1.fc39.noarch.rpm: Curl error (28): Timeout was reached for http://fedora-archive.ip-connect.vn.ua/f
edora/linux/updates/39/Everything/x86_64/Packages/p/pnpm-8.12.0-1.fc39.noarch.rpm [Failed to connect to fedora-archive.ip
-connect.vn.ua port 80 after 30002 ms: Timeout was reached]
pnpm-8.12.0-1.fc39.noarch.rpm                               7.7 kB/s | 2.6 MB    00:26

```

```

jacques@vbox:~$ pnpm setup
Appended new lines to /home/jacques/.bashrc

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

To start using pnpm, run:
source /home/jacques/.bashrc
jacques@vbox:~$ source /home/jacques/.bashrc

```

```
jacques@vbox:~$ pnpm add -g commitizen
```

Update available! 8.12.0 → 10.15.0.
Changelog: <https://github.com/pnpm/pnpm/releases/tag/v10.15.0>
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
```

```
/home/jacques/.local/share/pnpm/global/5:
```

```
+ commitizen 4.3.1
```

```
Done in 40.3s
```

```
jacques@vbox:~$ pnpm add -g standard-changelog
```

```
WARN 2 deprecated subdependencies found: glob@7.2.3, inflight@1.0.6
```

```
Packages: +34
```

```
Progress: resolved 185, reused 151, downloaded 34, added 34, done
```

```
/home/jacques/.local/share/pnpm/global/5:
```

```
+ standard-changelog 7.0.1
```

```
Done in 18.7s
```

```
jacques@vbox:~$
```

```
jacques@vbox:~$ git init
```

```
Dépôt Git vide initialisé dans /home/jacques/.git/
```

```
jacques@vbox:~$ git remote add origin git@github.com:Jacques-003/git-extended.git
```

```
jacques@vbox:~$ echo "# Git Extended Repo" > README.md
```

```
jacques@vbox:~$ git add .
```

```
jacques@vbox:~$ pnpm init
```

```
Wrote to /home/jacques/package.json
```

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

```
jacques@vbox:~$
```

```

jacques@vbox:~$ standard-changelog --first-release
file:///home/jacques/.local/share/pnpm/global/5/.pnpm/@simple-libs+child-process-utils@1.0.1/node_modules/@simple-libs/child-process-utils/dist/index.js:33
    error = new Error(stderr);
              ^
Error: fatal : argument 'HEAD' ambigu : révision inconnue ou chemin inexistant.
Utilisez '--' pour séparer les chemins des révisions, comme ceci :
'git <commande> [<révision>...] -- [<chemin>...]'

    at catchProcessError (file:///home/jacques/.local/share/pnpm/global/5/.pnpm/@simple-libs+child-process-utils@1.0.1/node_modules/@simple-libs/child-process-utils/dist/index.js:33:17)
    at async outputStream (file:///home/jacques/.local/share/pnpm/global/5/.pnpm/@simple-libs+child-process-utils@1.0.1/node_modules/@simple-libs/child-process-utils/dist/index.js:66:19)
    at async ConventionalGitClient.getTags (file:///home/jacques/.local/share/pnpm/global/5/.pnpm/@conventional-changelog+git-client@2.5.1/conventional-commits-parser@6.2.0/node_modules/@conventional-changelog/git-client/dist/GitClient.js:91:20)
    at async ConventionalGitClient.getSemverTags (file:///home/jacques/.local/share/pnpm/global/5/.pnpm/@conventional-changelog+git-client@2.5.1/conventional-commits-parser@6.2.0/node_modules/@conventional-changelog/git-client/dist/ConventionalGitClient.js:71:20)
    at async StandardChangelog.getSemverTags (file:///home/jacques/.local/share/pnpm/global/5/.pnpm/conventional-changelog@7.1.1/node_modules/conventional-changelog/dist/ConventionalChangelog.js:166:26)
    at async StandardChangelog.write (file:///home/jacques/.local/share/pnpm/global/5/.pnpm/conventional-changelog@7.1.1/node_modules/conventional-changelog/dist/ConventionalChangelog.js:412:28)
    at async runProgram (file:///home/jacques/.local/share/pnpm/global/5/.pnpm/conventional-changelog@7.1.1/node_modules/conventional-changelog/dist/cli/cli.js:156:26)
    at async file:///home/jacques/.local/share/pnpm/global/5/.pnpm/standard-changelog@7.0.1/node_modules/standard-changelog/dist/cli.js:35:1

Node.js v20.17.0
jacques@vbox:~$ git add CHANGELOG.md
jacques@vbox:~$ git commit -am "chore(site): add changelog"
erreur : gpg n'a pas pu signer les données :
gpg: « Jacques <nyemeckya@jacques@gmail.com> » a été ignorée : Pas de clef secrète
[GNUPG:] INV_SGNR 9 Jacques <nyemeckya@jacques@gmail.com>
[GNUPG:] FAILURE sign 17
gpg: signing failed: Pas de clef secrète

```