Отчет по лабораторной работе №4

Операционные системы

Мустафина Аделя Юрисовна

Содержание

Список иллюстраций

Список таблиц

# 1 Цель работы

Получение навыков правильной работы с репозиториями git.

# 2 Задание

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

# 3 Теоретическое введение

### 3.0.1 Общая информация

* 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 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

## 4.1 Установка программного обеспечения

Устанавливаю git-flow (рис. 1).

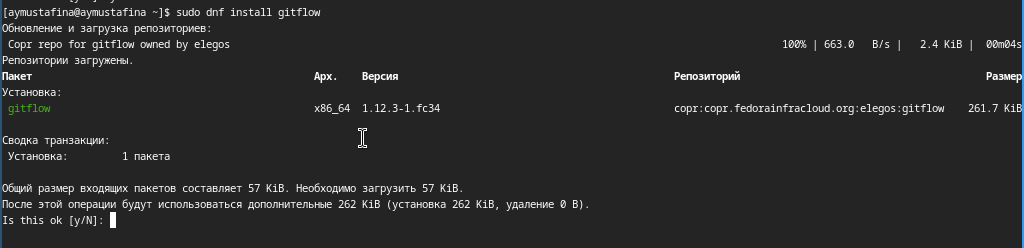


Рис. 1: Установка git-flow

Устанавливаю Node.js На Node.js базируется программное обеспечение для семантического версионирования и >

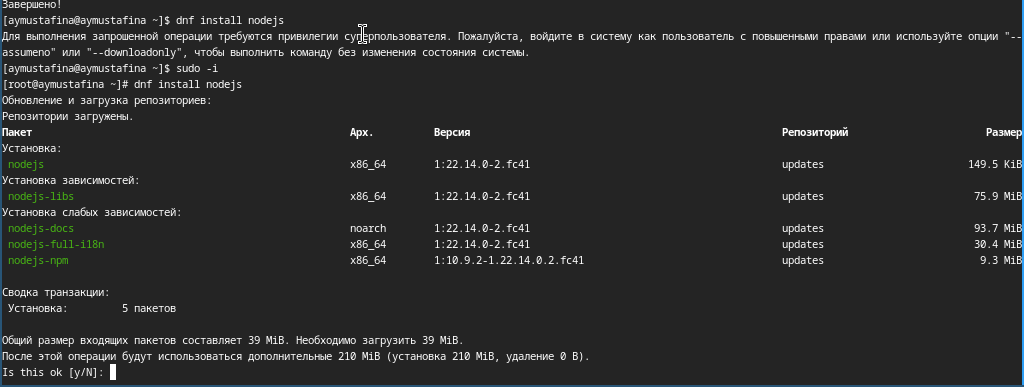


Рис. 2: Установка Node.js

Устанавливаю pnpm (рис. 3).

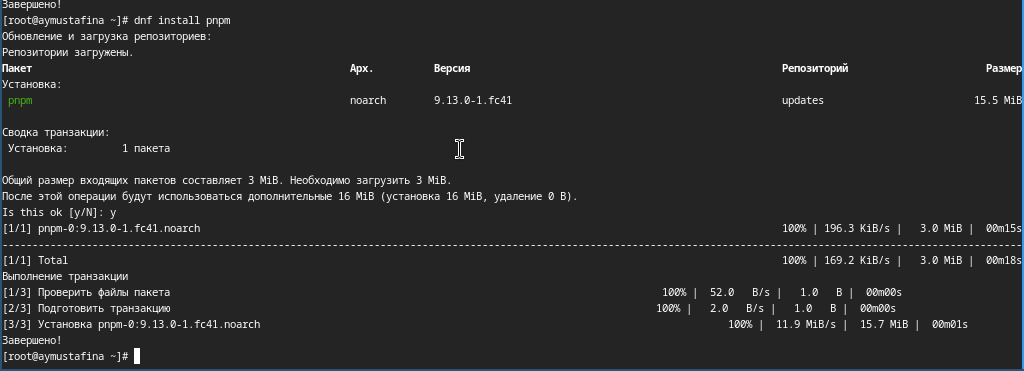


Рис. 3: Установка pnpm

Настраиваю Node.js. Для работы с Node.js добавляю каталог с исполняемыми файлами, устанавливаемыми yarn, в переменную PATH.(рис. 3).

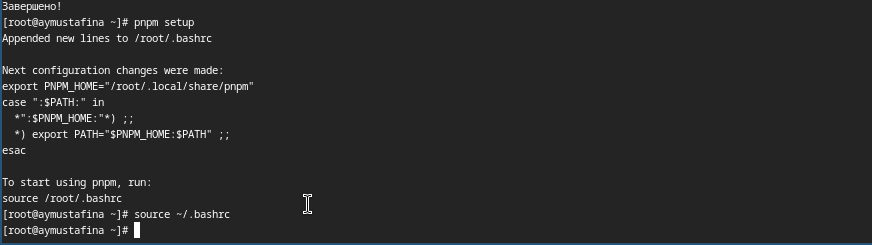


Рис. 4: Настройка

Устанавливаю commitizen(рис. 5).

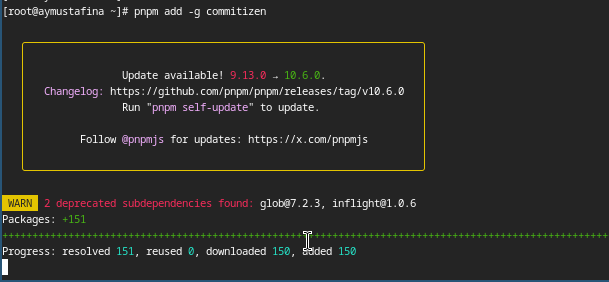


Рис. 5: Установка commitizen

Устанавливаю standard-changelog(рис. 6).

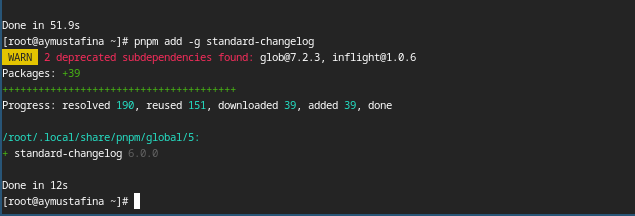


Рис. 6: Установка standard-changelog

Создаю новый репозиторий на github, называю его git-extended(рис. 7).

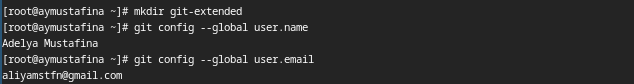


Рис. 7: Создание репозитория

Делаю первый коммит (рис. 8).

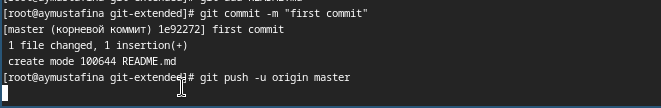


Рис. 8: Первый коммит

Выкладываю его на github (рис. 9).

Рис. 9: Выкладываю

Рис. 9: Выкладываю

Конфигурация для пакетов Node.js (рис. 10).

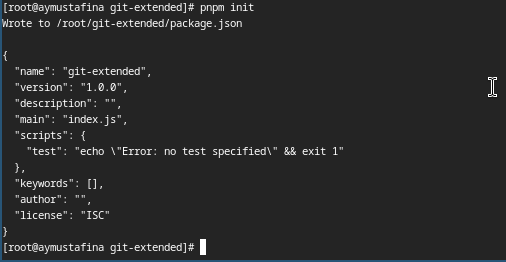


Рис. 10: Конфигурация

Открываю в редакторе и заполняю несколько параметров пакета. аким 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"  
 }  
 }   
}

(рис. 11).

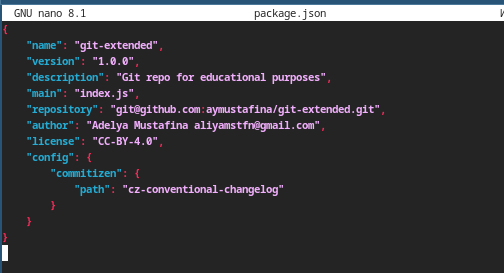


Рис. 11: Файл package.json

Добавляю новые файлы. Выполняю коммит (рис. 12).

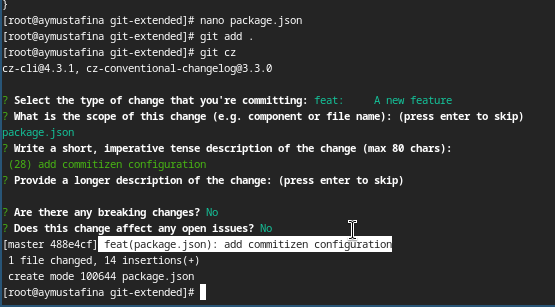


Рис. 12: Коммит

Отправляю все на github (рис. 13).

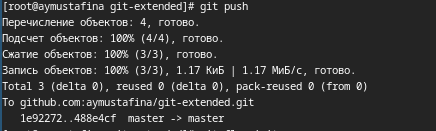


Рис. 13: Отправка

Инициализирую git-flow, префикс для ярлыков устанавливаю в v. Проверяю, что я на ветке develop. Загружаю все в хранилище (рис. 14).

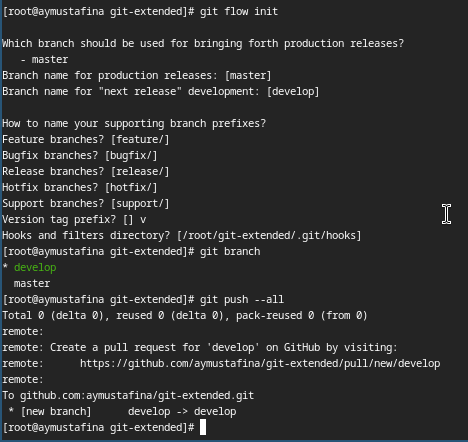


Рис. 14: Иницаиализация

Устанавливаю внешнюю ветку как вышестоящую для этой ветки и создаю релиз с версией 1.0.0 (рис. 15).

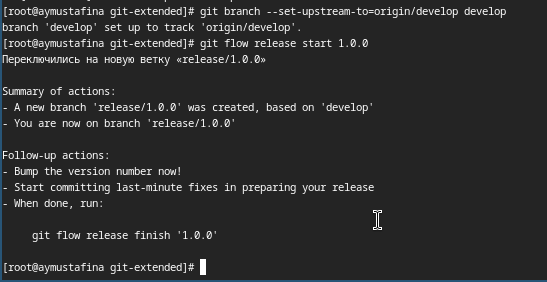


Рис. 15: Релиз 1.0.0

Создаю журнал изменений standard-changelog –first-release (рис. 16).

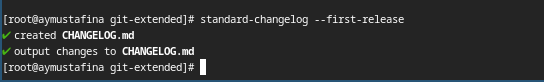


Рис. 16: Журнал изменений

Добавляю журнал изменений в индекс (рис. 17).

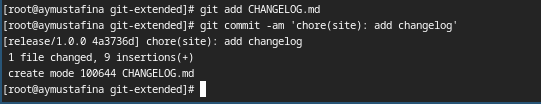


Рис. 17: Журнал изменений в индекс

Заливаю релизную ветку в основную ветку (рис. 18).

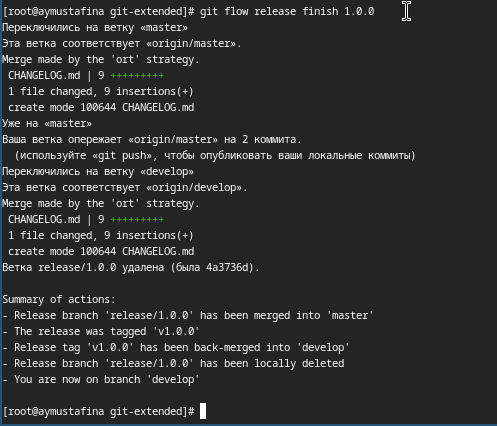


Рис. 18: Релизная в основную

Отправляю данные на github и создаю релиз (рис. 19).

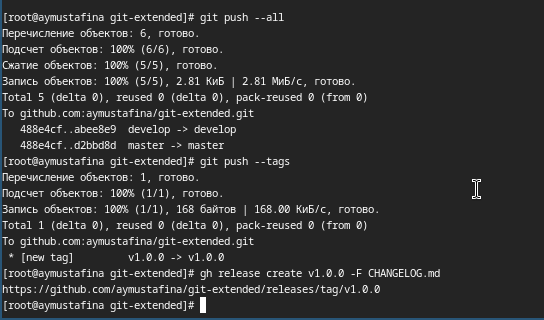


Рис. 19: Отправка и релиз

Создаю ветку для новой функциональности и по окончании работы с git объединяю ветку feature\_branch c develop (рис. 20).

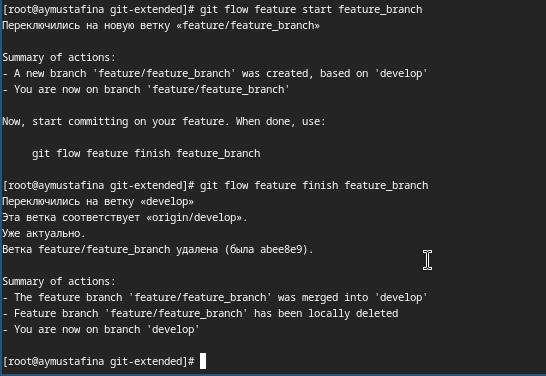


Рис. 20: Создание и объединение веток

Создаю релиз с версией 1.2.3, в файле обновляю ее номер (рис. 21).

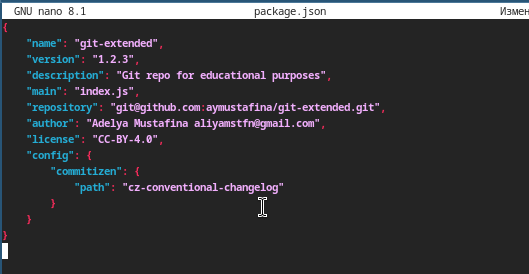


Рис. 21: Релиз 1.2.3

Создаю журнал изменений (рис. 22).

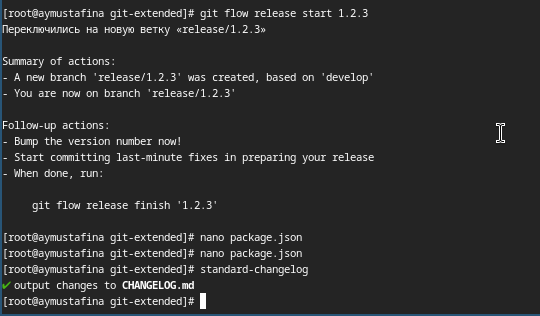


Рис. 22: Журнал изменений

Добавляю журнал изменений в индекс и заливаю релизную ветку в основную (рис. 23).

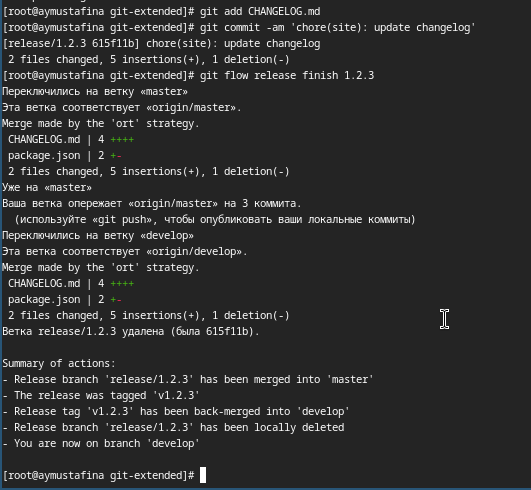


Рис. 23: Журнал в индекс

Отправляю данные на гитхаб и создаю релиз с комментарием из журнала изменений (рис. 24).

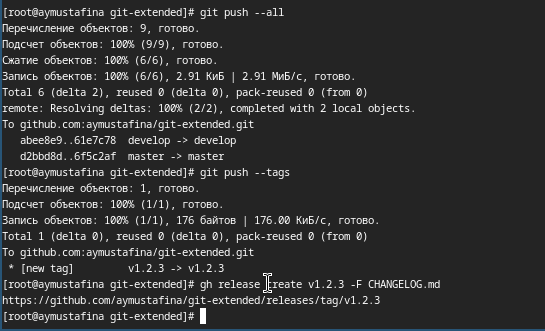


Рис. 24: Отправка

# 5 Выводы

Я научилась работать с релизами в Github, узнала юольше о типах и структуре коммитов.

# 6 Список литературы

1. Лабораторная работа №4 [1]

1. [Лабораторная работа №4](https://esystem.rudn.ru/mod/page/view.php?id=1224375 ).