Отчёт по лабораторной работе №4

Операционные системы

Пономарева Татьяна Александровна

Содержание

# 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

Рабочий процесс Gitflow Workflow используется с пакетом git-flow. Gitflow Workflow опубликована и популяризована Винсентом Дриссеном и предполагает выстраивание строгой модели ветвления с учетом выпуска проекта.

# 4 Выполнение лабораторной работы

## 4.1 Установка Node.js

Сначала устанавливаю Node.js и pnpm (рис. 1).

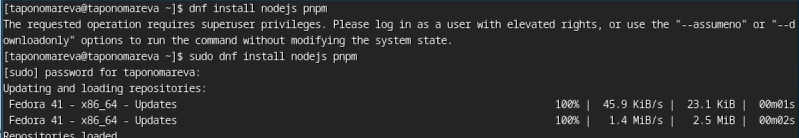


Рис. 1: Установка nodejs и pnpm

## 4.2 Установка gitflow

Потом загружаю gitflow (рис. 2).

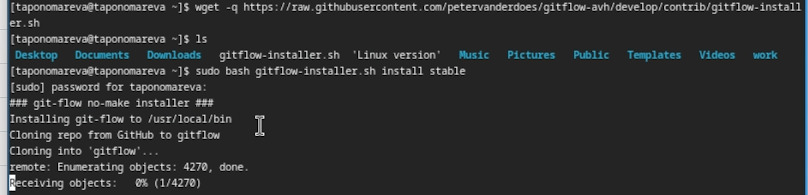


Рис. 2: Установка gitflow

## 4.3 Настройка Node.js

Далее идет настройка Node.js. Запускаю pnpm, делаю перелогин и выполняю source ~/.bashrc (рис. 3).

Настройка Node.js

Рис. 3: Настройка Node.js

## 4.4 Общепринятые коммиты

Использую commitizen для форматирования коммитов (рис. 4).

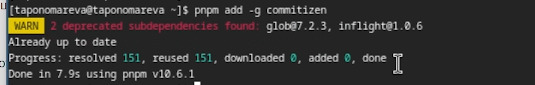


Рис. 4: Commitizen

Использую standard-changelog для создания логов (рис. 5).

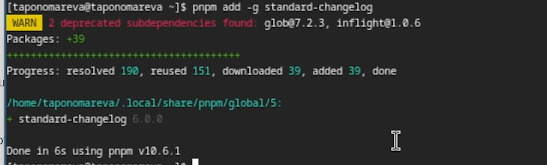


Рис. 5: Standard-changelog

## 4.5 Создание репозитория git

Создаю репозиторий git-extended и копирую его в папку home (рис. 6).

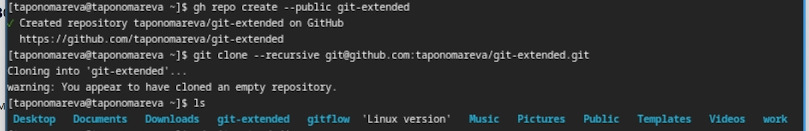


Рис. 6: Создание git-extended

Делаю первый коммит и выкладываю его на github (рис. 7).

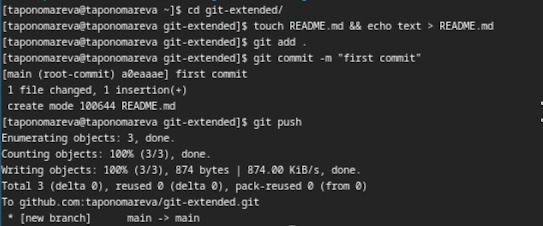


Рис. 7: Создание коммита и выкладывание на github

## 4.6 Конфигурация общепринятых коммитов

Использую команду pnpm init, затем nano package.json (рис. 8)

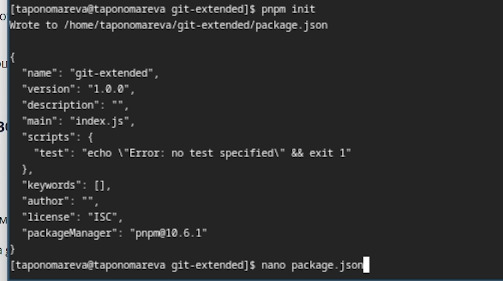


Рис. 8: Конфигурация для пакетов Node.js

Изменяю файл package.json (рис. 9).

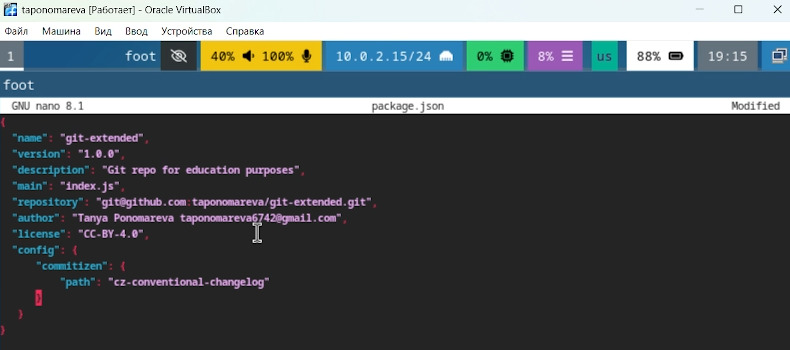


Рис. 9: package.json

Добавляю новые файлы, выполняю коммит и отправляю на github (рис. 10).

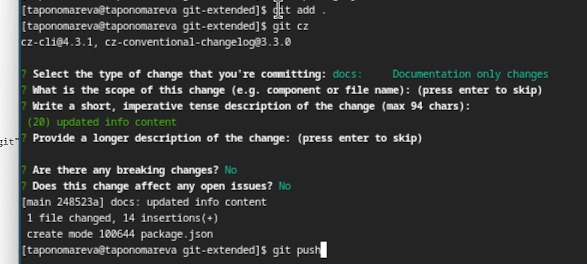


Рис. 10: Отправление на github

## 4.7 Конфигурация git-flow

Инициализирую git-flow, устанавливаю префикс для ярлыков как v, проверяю, что нахожусь на ветке develop (рис. 11).

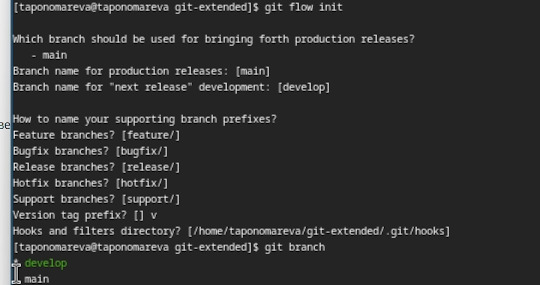


Рис. 11: Работа с git-flow

Загружаю весь репозиторий в хранилище и устанавливаю внешнюю ветку как вышестоящую для этой ветки (рис. 12).

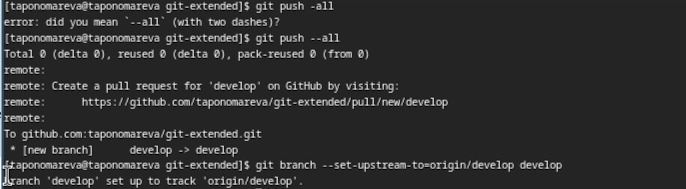


Рис. 12: Загрузка репозитория и установка внешней ветки

Создаю релиз с версией 1.0.0 (рис. 13).

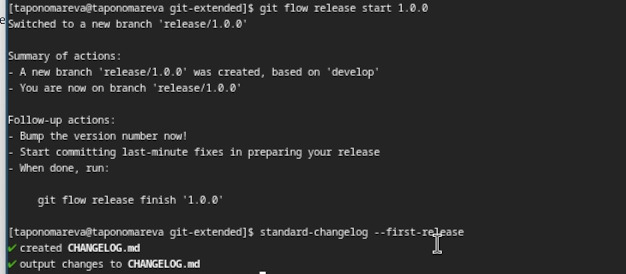


Рис. 13: Создание релиза с версией 1.0.0

Создаю журнал изменений и добавляю его в индекс(рис. 14).

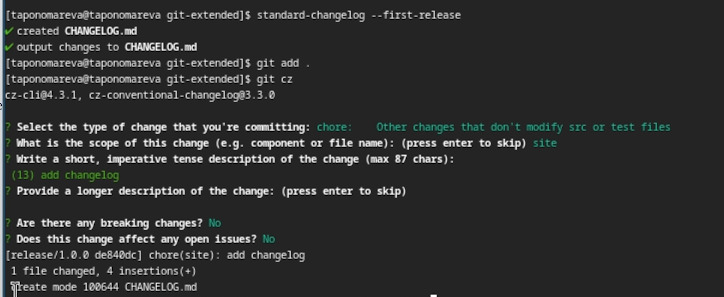


Рис. 14: Создание журнала изменений и его добавление в индекс

Заливаю релизную ветку в основную (рис. 15).

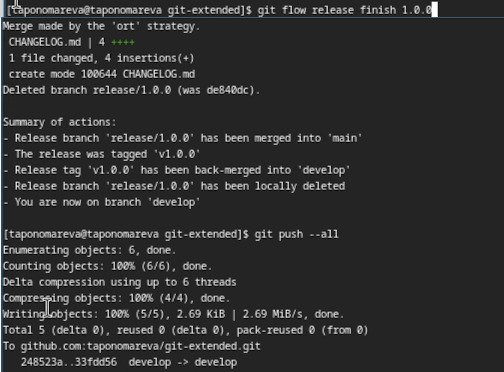


Рис. 15: Релизная ветка

Отправляю данные на github (рис. 16).

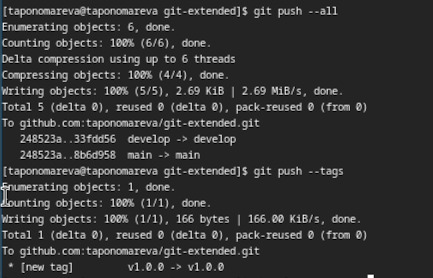


Рис. 16: Отправка данных на github

Создаю релиз на гитхаб (рис. 17).

Создание релиза на гитхабе

Рис. 17: Создание релиза на гитхабе

Создаю ветку для новой функциональности, затем создаю релиз 1.2.3 (рис. 18).

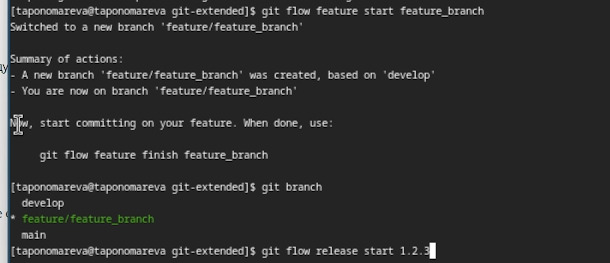


Рис. 18: Создание новой ветки

Обновляю номер версии в файле package.json (рис. 19).

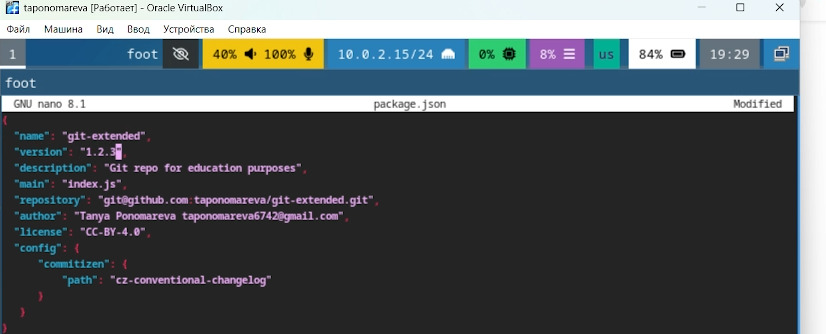


Рис. 19: Файл package.json

Создаю журнал изменений, добавляю его в индекс и заливаю релизную ветку в основную (рис. 20).

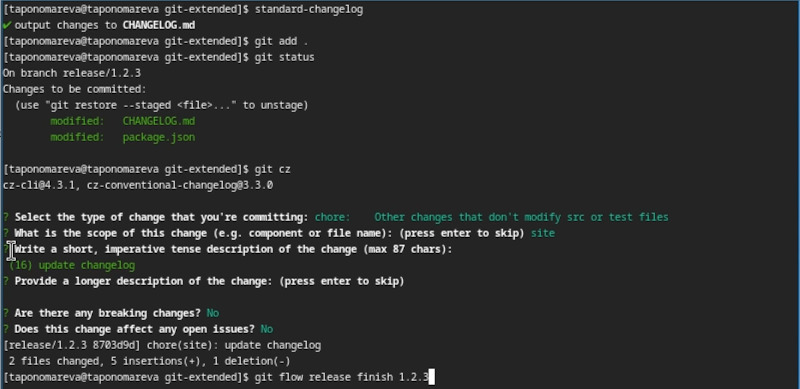


Рис. 20: Создание релиза git-flow

Отправляю данные на гитхаб (рис. 21).

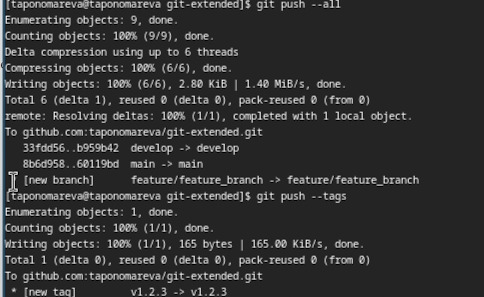


Рис. 21: Отправка данных на гитхаб

Создаю релиз на гитхаб с комментарием из журнала изменений (рис. 22).

Создание релиза на гитхабе с комментарием из журнала изменений

Рис. 22: Создание релиза на гитхабе с комментарием из журнала изменений

# 5 Выводы

Были получены навыков правильной работы с репозиториями git

# Список литературы

1. [Курс на ТУИС](https://esystem.rudn.ru/course/view.php?id=113)