Лабораторная работа №4

Отчёт

Александр Денисович Мосолов

Содержание

# 1 Цель работы

Получение навыков правильной работы с репозиториями git.[1]

# 2 Задание

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

# 3 Выполнение лабораторной работы

Установка из коллекции репозиториев Copr (рис. 1):

Enable the copr repository

Рис. 1: Enable the copr repository

Установка gitflow (рис. 2):

Install gitflow

Рис. 2: Install gitflow

Установка Node.js (рис. 3):

Установка Node.js

Рис. 3: Установка Node.js

Установка pnpm с помощью wget(рис. 4):

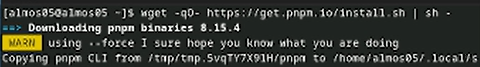


Рис. 4: Установка pnpm

Запускаем pnpm и выполняем (рис. 5):

Проверка

Рис. 5: Проверка

Создаем пустой репозиторий и клонируем его (рис. 6):

Клонируем репозиторий

Рис. 6: Клонируем репозиторий

standard-changelog используется для помощи в создании логов (рис. 7):



Рис. 7: Standard-changelog

Commitizen используется для помощи в форматировании коммитов (рис. 8):

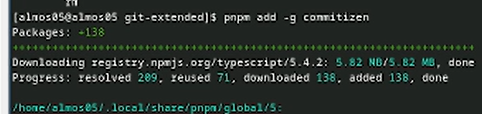


Рис. 8: Commitizen

Создаем пустой файл README.md, фиксируем изменения и делаем первый коммит (рис. 9):

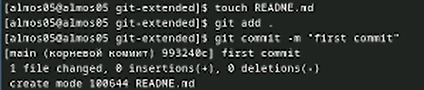


Рис. 9: Первый коммит

Делаем git push в ветку main (рис. 10):

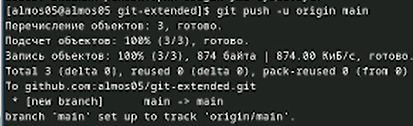


Рис. 10: git push в ветку main

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



Рис. 11: pnpm init

Изменяем package.json (рис. 12):



Рис. 12: Изменяем package.json

Добавим новые файлы:  
git add .

Выполним коммит:  
git cz

Отправим на github:  
git push

Инициализируем git-flow, префикс для ярлыков установим в v (рис. 13):

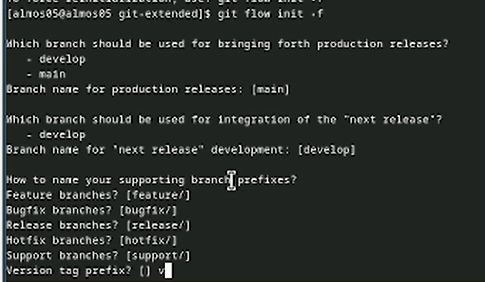


Рис. 13: Инициализируем git-flow

Проверьте, что Вы на ветке develop (рис. 14):

Ветка develop

Рис. 14: Ветка develop

Загружаем весь репозиторий в хранилище (рис. 15):

git push –all

Рис. 15: git push –all

Установим внешнюю ветку как вышестоящую для этой ветки (рис. 16):

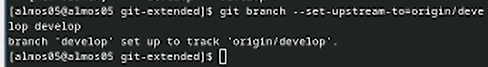


Рис. 16: Внешняя ветка становится вышестоящей

Создадим релиз с версией 1.0.0 (рис. 17):

Создадим релиз с версией 1.0.0

Рис. 17: Создадим релиз с версией 1.0.0

Создадим журнал изменений (рис. 18):  
standard-changelog –first-release  
И добавим журнал изменений в индекс

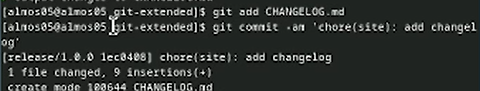


Рис. 18: Добавим журнал изменений в индекс

Зальём релизную ветку в основную ветку  
git flow release finish 1.0.0  
Отправим данные на github (рис. 19):

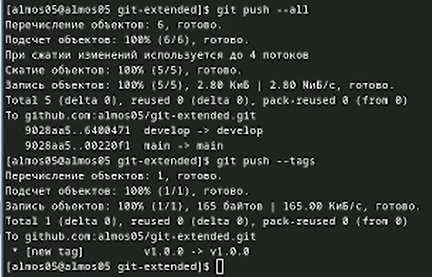


Рис. 19: Отправим данные на github

Создадим релиз на github. Для этого будем использовать утилиты работы с github (рис. 20):

Создадим релиз на github

Рис. 20: Создадим релиз на github

Создадим ветку для новой функциональности (рис. 21):

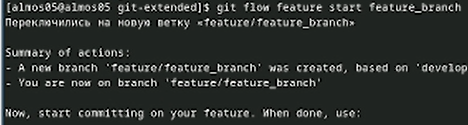


Рис. 21: Создадим ветку для новой функциональности

Создадим релиз с версией 1.2.3 (рис. 22):

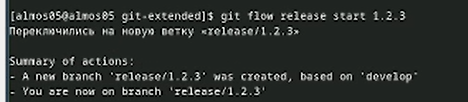


Рис. 22: Создадим релиз с версией 1.2.3

Обновите номер версии в файле package.json. Установите её в 1.2.3 (рис. 23):

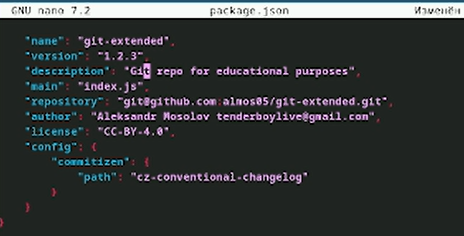


Рис. 23: Обновите номер версии в файле package.json

Создадим журнал изменений (рис. 24):

Создадим журнал изменений

Рис. 24: Создадим журнал изменений

Добавим журнал изменений в индекс  
git add CHANGELOG.md, сделаем новый коммит(рис. 25):

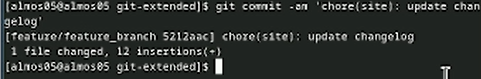


Рис. 25: Добавим журнал изменений в индекс

Зальём релизную ветку в основную ветку (рис. 26):

Зальём релизную ветку в основную ветку

Рис. 26: Зальём релизную ветку в основную ветку

Отправим данные на github (рис. 27):

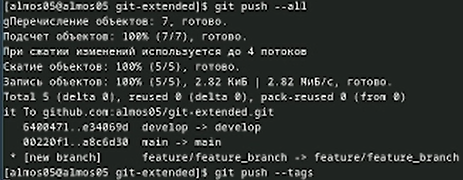


Рис. 27: Отправим данные на github

Зальём релизную ветку в основную Создадим релиз на github с комментарием из журнала изменений]):

Создадим релиз на github с комментарием из журнала изменений

Рис. 28: Создадим релиз на github с комментарием из журнала изменений

# 4 Выводы

В ходе работы мы получили навыки правильной работы с репозиториями git.

# Список литературы

1. Kulyabov. [Архитектура компьютеров и операционные системы. Раздел "Операционные системы" (09.03.03, НПИбд)](https://esystem.rudn.ru/mod/page/view.php?id=1098937).