Отчёт по лабораторной работе №4

Продвинутое использование git

Кылыч Гоктюрк

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

Получение навыков правильной работы с репозиториями git.

2 Теоретические сведения

- 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.

3 Выполнение лабораторной работы

3.1 Работа с тестовым репозиторием

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

Рисунок 3.1: Node.js

Программа commitizen используется для помощи в форматировании коммитов. При этом устанавливается скрипт git-cz, который мы и будем использовать для коммитов.

Рисунок 3.2: установка commitizen

Программа standard-changelog используется для помощи в создании логов.
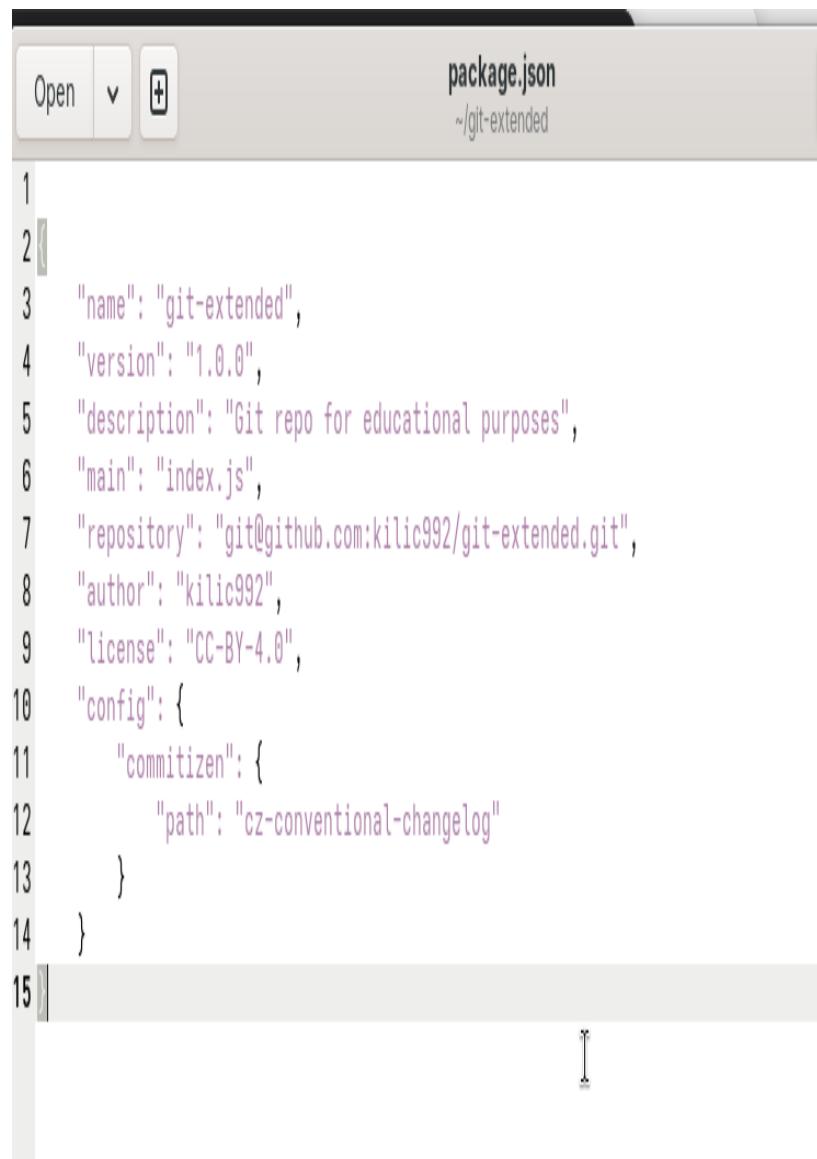
Рисунок 3.3: установка standard-changelog

Делаем первый коммит и выкладываем на github.

Необходимо заполнить несколько параметров пакета.

Таким образом, файл package.json приобретает вид:

Рисунок 3.4: package.json

Добавим новые файлы.

Выполним коммит.

Отправим на github.

Рисунок 3.5: Отправка

Инициализируем git-flow

Проверьте, что Вы на ветке develop

Загрузите весь репозиторий в хранилище

Рисунок 3.6: Инициализация и загрузка изменений

Установите внешнюю ветку как вышестоящую для этой ветки

Создадим релиз с версией 1.0.0

Создадим журнал изменений

Добавим журнал изменений в индекс

Зальём релизную ветку в основную ветку

Рисунок 3.7: Завершение релиза

Отправим данные на github

Рисунок 3.8: Отправка

Создадим ветку для новой функциональности По окончании разработки новой функциональности следующим шагом следует объединить ветку feature_branch с develop:

Рисунок 3.9: Объединение веток

Создадим релиз с версией 1.2.3

Обновите номер версии в файле package.json. Установите её в 1.2.3

Создадим журнал изменений

Добавим журнал изменений в индекс

Зальём релизную ветку в основную ветку

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

kilicgokturk@kilicgokturk:~/git-extended$ git push --all
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 2.77 KiB | 2.77 MiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:kilic992/git-extended.git
   2ee8627..e731f65  develop -> develop
   cf334f0..f674145  master -> master
kilicgokturk@kilicgokturk:~/git-extended$ git push --tags
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 154 bytes | 154.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:kilic992/git-extended.git
 * [new tag]       v1.2.3 -> v1.2.3
kilicgokturk@kilicgokturk:~/git-extended$ gh release create v1.2.3 -F CHANGELOG.md
https://github.com/kilic992/git-extended/releases/tag/v1.2.3
kilicgokturk@kilicgokturk:~/git-extended$
```

Рисунок 3.10: Завершение релиза

3.2 Подготовка рабочего репозитория

Рисунок 3.11: package.json и коммит

```
Summary of actions:
- Release branch 'release/1.0.0' has been merged into 'master'
- The release was tagged 'v1.0.0'
- Release tag 'v1.0.0' has been back-merged into 'develop'
- Release branch 'release/1.0.0' has been locally deleted
- You are now on branch 'develop'


kilicgokturk@kilicgokturk:~/work/study/2025-2026/Операционные системы/2026-1--study--os-intro$ git p
ush --all
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 2.74 KiB | 2.74 MiB/s, done.
Total 5 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:kilic992/2026-1--study--os-intro.git
   6694626..e55114b  develop -> develop
   6694626..8805bd5  master -> master
kilicgokturk@kilicgokturk:~/work/study/2025-2026/Операционные системы/2026-1--study--os-intro$ git p
ush --tags
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 154 bytes | 154.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:kilic992/2026-1--study--os-intro.git
 * [new tag]         v1.0.0 -> v1.0.0
kilicgokturk@kilicgokturk:~/work/study/2025-2026/Операционные системы/2026-1--study--os-intro$
```

Рисунок 3.12: Завершение релиза

4 Вывод

Мы приобрели практические навыки взаимодействия с дополнительными функциями гитхаб.