# Oтчёта по лабораторной работе 4

НММ-03

Чжан Сюйтун

|  |  |
| --- | --- |
| **Содержание** |  |
| **[Цель работы](#_bookmark0)** | **5** |
| **[Задание](#_bookmark1)** | **6** |
| **[Лабораторной работы](#_bookmark2)** | **7** |
| [Общепринятые коммиты](#_bookmark6) | . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8 |
| **[Выводы](#_bookmark19)** | **17** |
| **[Список литературы](#_bookmark20)** | **18** |

# Список иллюстраций

1. [Установка git-flow](#_bookmark3) 7
2. [Установка Node.js](#_bookmark4) 8
3. [Настройка Node.js](#_bookmark5) 8
4. [Commitizen](#_bookmark7) 9
5. [standard-changelog](#_bookmark8) 9
6. [git-extended](#_bookmark9) 10
7. [первый коммит](#_bookmark10) 10
8. [файл package.json](#_bookmark11) 11
9. [Git](#_bookmark12) 11
10. [Конфигурация git-flow 1](#_bookmark13) 12
11. [Конфигурация git-flow 2](#_bookmark14) 13
12. [Конфигурация git-flow 3](#_bookmark15) 14
13. [Разработка новой функциональности](#_bookmark16) 15
14. [Создание релиза git-flow](#_bookmark17) 16
15. [github](#_bookmark18) 16

**Список таблиц**

# 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.

# Лабораторной работы

Установка git-flow из коллекции репoзитoриев Copr (https://copr.fedorainfracloud.org/coprs/elegos/

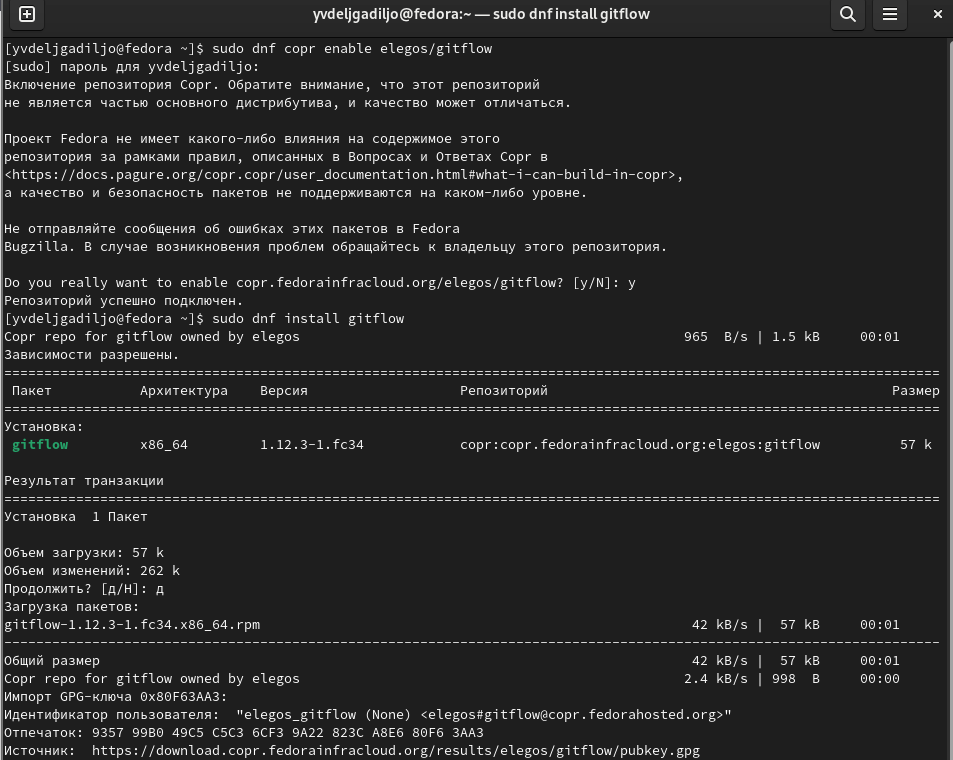


Рис. 1: Установка git-flow

Установка Node.js. На Node.js базируется программное обеспечение для семанти- ческого версионирования и общепринятых коммитов.

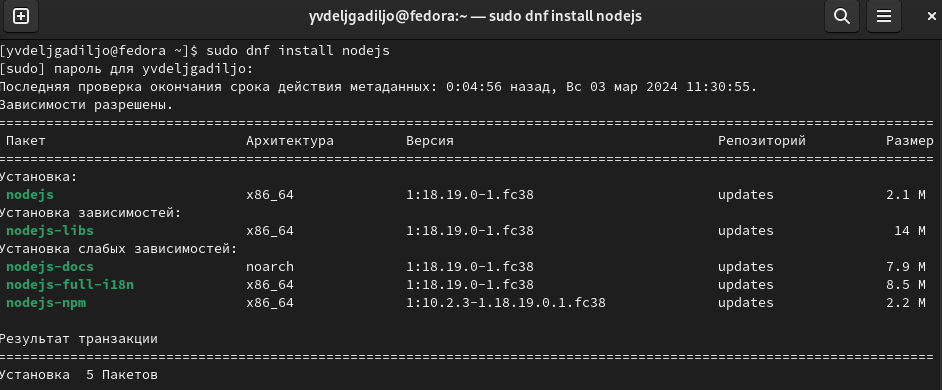


Рис. 2: Установка Node.js

Для работы с Node.js добавим каталог с исполняемыми файлами, устанавливаемы- ми yarn, в переменную PATH.

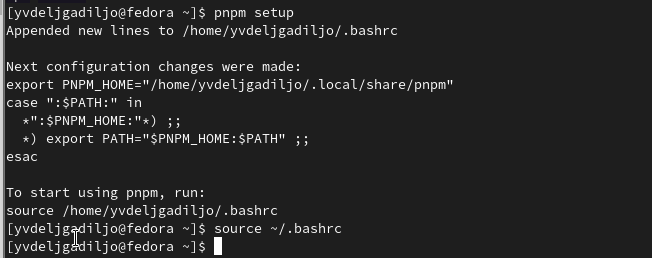


Рис. 3: Настройка Node.js

**Общепринятые коммиты**

Данная программа используется для помощи в форматировании коммитов.

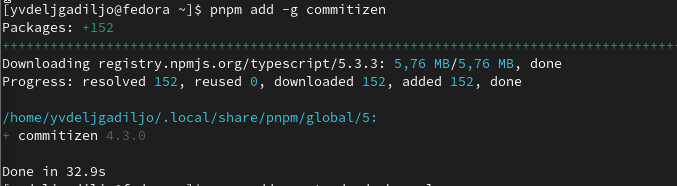


Рис. 4: Commitizen

Данная программа используется для помощи в создании логов.

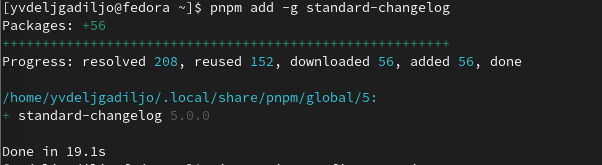


Рис. 5: standard-changelog

Cоздание репозитория git

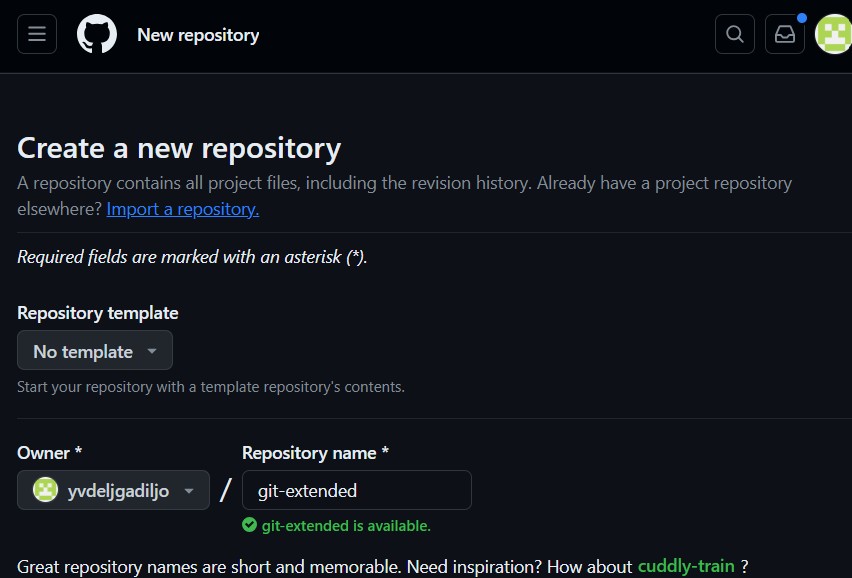


Рис. 6: git-extended

Делаем первый коммит и выкладываем на github:

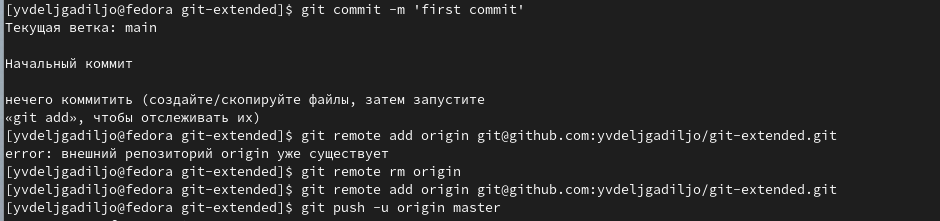


Рис. 7: первый коммит

Сконфигурим формат коммитов. Для этого добавим в файл package.json команду для формирования коммитов:

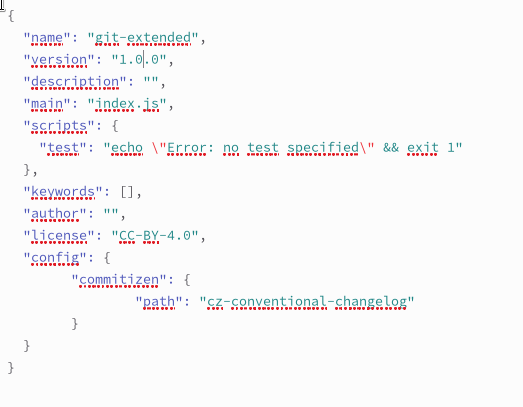


Рис. 8: файл package.json

Добавим новые файлы, выполним коммит и oтправим на github

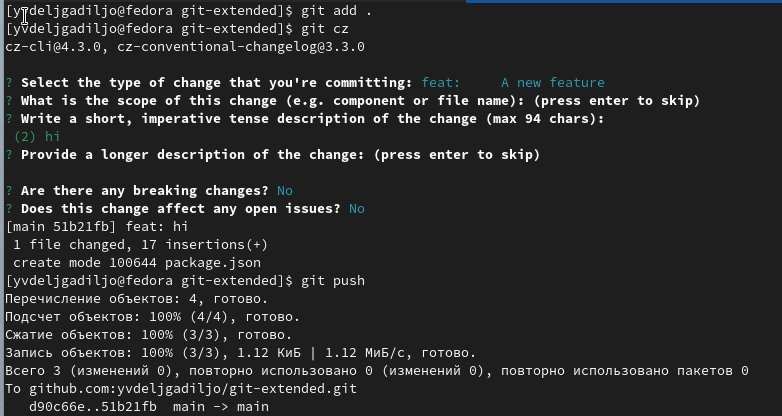


Рис. 9: Git

Инициализируем git-flow, проверьте, что Вы на ветке develop и загрузите весь ре- позиторий в хранилище.

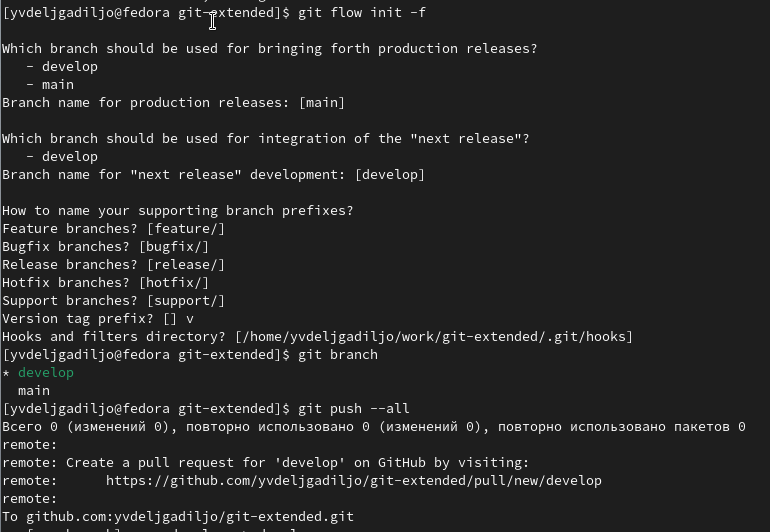


Рис. 10: Конфигурация git-flow 1

Установите внешнюю ветку как вышестоящую для этой ветки, создадим релиз с вер- сией 1.0.0, создадим журнал изменений и добавим журнал изменений в индекс.

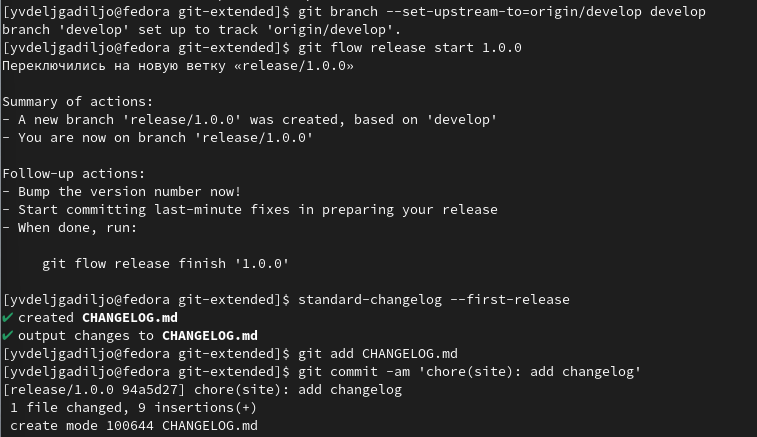


Рис. 11: Конфигурация git-flow 2

Зальём релизную ветку в основную ветку, отправим данные на github и создадим релиз на github. Для этого будем использовать утилиты работы с github:

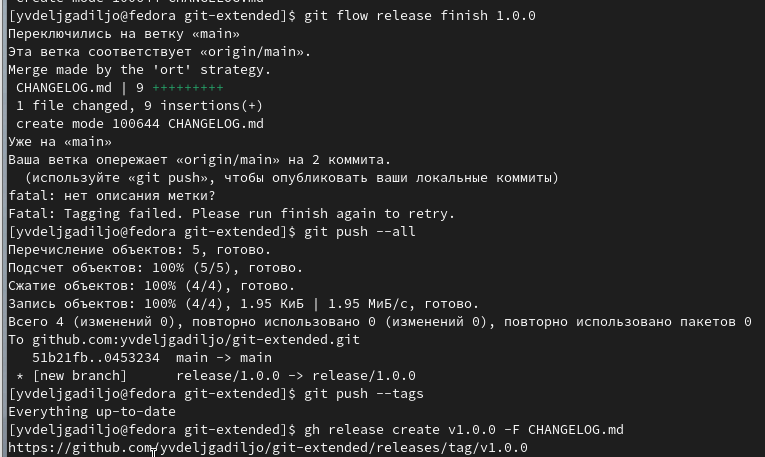


Рис. 12: Конфигурация git-flow 3

Создадим ветку для новой функциональности и по окончании разработки новой функциональности следующим шагом следует объединить ветку feature\_branch c develop.

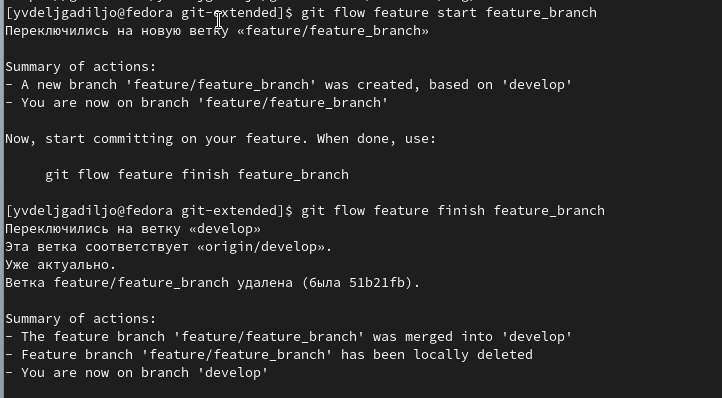


Рис. 13: Разработка новой функциональности

Создадим релиз с версией 1.2.3, обновите номер версии в файле package.json. Уста- новите её в 1.2.3. Создадим журнал изменений и добавим журнал изменений в индекс и зальём релизную ветку в основную ветку.

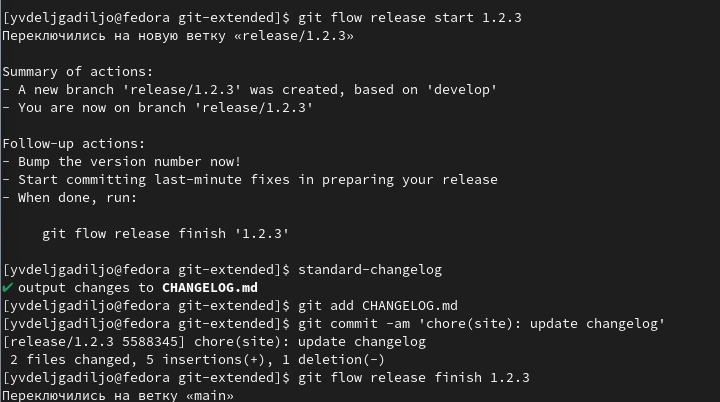


Рис. 14: Создание релиза git-flow

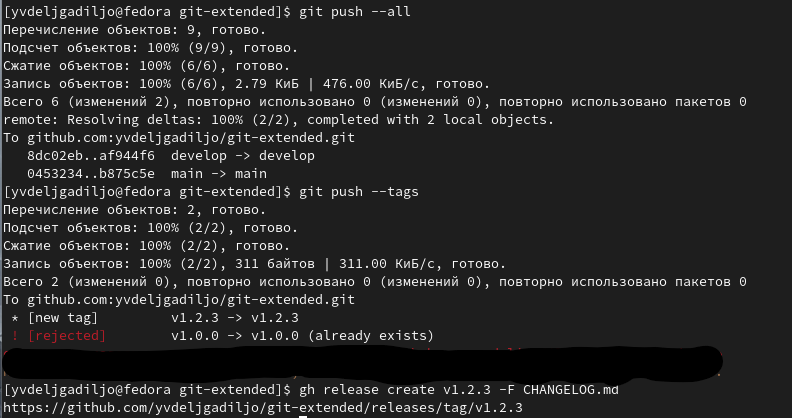
Отправим данные на github и создадим релиз на github с комментарием из журнала изменений:

Рис. 15: github

# Выводы

* Освоила умения по работе с git.
* Приобрелa навыки правильной работы с git-репозиториям.

# Список литературы

* GDB: The GNU Project Debugger. — URL: https:[//www.gnu.org/so](http://www.gnu.org/software/gdb/)f[tware/gdb/.](http://www.gnu.org/software/gdb/)
* GNU Bash Manual. — 2016. — URL: https:[//www.gnu.org/so](http://www.gnu.org/software/bash/manual/)f[tware/bash/manual/.](http://www.gnu.org/software/bash/manual/)
* Midnight Commander Development Center. — 2021. — URL: https://midnight- commander.org/.
* NASM Assembly Language Tutorials. — 2021. — URL: https://asmtutor.com/.
* Newham C. Learning the bash Shell: Unix Shell Programming. — O’Reilly Media, 2005. —354 с. — (In a Nutshell). — ISBN 0596009658. — URL: [http://www.amazon.com/Learningbash-Shell-Programming-Nutshell/dp/0596009658.](http://www.amazon.com/Learningbash-Shell-Programming-Nutshell/dp/0596009658)
* Robbins A. Bash Pocket Reference. — O’Reilly Media, 2016. — 156 с. — ISBN 978- 1491941591.
* The NASM documentation. — 2021. — URL: https://[www.nasm.us/docs.php.](http://www.nasm.us/docs.php)
* Zarrelli G. Mastering Bash. — Packt Publishing, 2017. — 502 с. — ISBN 9781784396879.
* Колдаев В. Д., Лупин С. А. Архитектура ЭВМ. — М. : Форум, 2018.
* Куляс О. Л., Никитин К. А. Курс программирования на ASSEMBLER. — М. : Солон- Пресс, 2017.
* Новожилов О. П. Архитектура ЭВМ и систем. — М. : Юрайт, 2016.
* Расширенный ассемблер: NASM. — 2021. — URL: https:[//www.opennet.ru/docs/RUS/nasm/.](http://www.opennet.ru/docs/RUS/nasm/)
* Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд.

— БХВПетербург, 2010. — 656 с. — ISBN 978-5-94157-538-1.

* Столяров А. Программирование на языке ассемблера NASM для ОС Unix. — 2-е изд. — М. : МАКС Пресс, 2011. — URL: [http://www.stolyarov.info/books/asm\_unix.](http://www.stolyarov.info/books/asm_unix)
* Таненбаум Э. Архитектура компьютера. — 6-е изд. — СПб. : Питер, 2013. — 874 с. — (Классика Computer Science).
* Таненбаум Э., Бос Х. Современные операционн