

Лабораторная работа 4

Продвинутое использование git

Головина Мария Игоревна

Содержание

1	Цель работы	6
2	Задание	7
3	Теоретическое введение	8
4	Выполнение лабораторной работы	14
5	Выводы	48
	Список литературы	49

Список иллюстраций

4.1	Репозиторий git-extended	14
4.2	Клонирование	15
4.3	# Enable the copr repository	16
4.4	dnf copr enable elegos/gitflow	17
4.5	# Install gitflow	18
4.6	dnf install gitflow	19
4.7	dnf install nodejs	20
4.8	dnf install pnpm	21
4.9	pnpm setup	22
4.10	source ~/.bashrc	23
4.11	pnpm add -g commitizen	24
4.12	pnpm add -g standard-changelog	25
4.13	git commit -m "first commit"	26
4.14	git remote add origin git@github.com:migolovina/git-extended.git . . .	26
4.15	git push -u origin main	27
4.16	package.json	27
4.17	Добавляем новые файлы, выполняем коммит и отправляем на github	28
4.18	Отправляем на github	28
4.19	Инициализируем git-flow	29
4.20	Проверка	30
4.21	Загрузка всего репозитория в хранилище	31
4.22	Установка внешней ветки как вышестоящую для этой ветки	32
4.23	Создание релиза с версией 1.0.0	33
4.24	Создание журнала изменений	34
4.25	Добавление журнала изменений в индекс	35
4.26	Добавление журнала изменений в индекс	36
4.27	Залитие релизной ветки в основную ветку	37
4.28	Отправление данных на github	38
4.29	Отправление данных на github	39
4.30	Создание релиза на github	40
4.31	Проверка	40
4.32	Создание ветки для новой функциональности	41
4.33	Объединение ветки feature_branch с develop	42
4.34	Создание релиза с версией 1.2.3:	43
4.35	Обновление номера версии в файле package.json.	43
4.36	Создание журнала изменений	44
4.37	Добавление журнала изменений в индекс	45

4.38 Залитие релизной ветки в основную ветку	46
4.39 Отправление данных на github и создание релиза на github с ком- ментарием из журнала изменений	47
4.40 Проверка релизов	47

Список таблиц

1 Цель работы

Получение навыков правильной работы с репозиториями git

2 Задание

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

3 Теоретическое введение

Рабочий процесс Gitflow

Рабочий процесс Gitflow Workflow. Будем описывать его с использованием пакета git-flow.

Общая информация

Gitflow Workflow опубликована и популяризована Винсентом Дриссенем. 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.

Процесс работы с Gitflow

Основные ветки (master) и ветки разработки (develop) Для фиксации истории проекта в рамках этого процесса вместо одной ветки master используются две ветки. В ветке master хранится официальная история релиза, а ветка develop предназначена для объединения всех функций. Кроме того, для удобства рекомендуется присваивать всем коммитам в ветке master номер версии.

При использовании библиотеки расширений git-flow нужно инициализировать

структуру в существующем репозитории:

```
git flow init
```

Для github параметр Version tag prefix следует установить в v.

После этого проверьте, на какой ветке Вы находитесь:

```
git branch
```

Функциональные ветки (feature) Под каждую новую функцию должна быть отведена собственная ветка, которую можно отправлять в центральный репозиторий для создания резервной копии или совместной работы команды. Ветки feature создаются не на основе master, а на основе develop. Когда работа над функцией завершается, соответствующая ветка сливается обратно с веткой develop. Функции не следует отправлять напрямую в ветку master. Как правило, ветки feature создаются на основе последней ветки develop.

Создание функциональной ветки

Создадим новую функциональную ветку:

```
git flow feature start feature_branch
```

Далее работаем как обычно.

Окончание работы с функциональной веткой

По завершении работы над функцией следует объединить ветку feature_branch с develop:

```
git flow feature finish feature_branch
```

Ветки выпуска (release) Когда в ветке develop оказывается достаточно функций для выпуска, из ветки develop создаётся ветка release. Создание этой ветки запускает следующий цикл выпуска, и с этого момента новые функции добавлять больше нельзя — допускается лишь отладка, создание документации и решение других задач. Когда подготовка релиза завершается, ветка release сливается с master и ей присваивается номер версии. После нужно выполнить слияние с веткой develop, в которой с момента создания ветки релиза могли возникнуть изменения. Благодаря тому, что для подготовки выпусков используется специальная ветка, одна команда может дорабатывать текущий выпуск, в то время как

другая команда продолжает работу над функциями для следующего.

Создать новую ветку release можно с помощью следующей команды:

```
git flow release start 1.0.0
```

Для завершения работы на ветке release используются следующие команды:

```
git flow release finish 1.0.0
```

Ветки исправления (hotfix) Ветки поддержки или ветки hotfix используются для быстрого внесения исправлений в рабочие релизы. Они создаются от ветки master. Это единственная ветка, которая должна быть создана непосредственно от master. Как только исправление завершено, ветку следует объединить с master и develop. Ветка master должна быть помечена обновлённым номером версии. Наличие специальной ветки для исправления ошибок позволяет команде решать проблемы, не прерывая остальную часть рабочего процесса и не ожидая следующего цикла релиза.

Ветку hotfix можно создать с помощью следующих команд:

```
git flow hotfix start hotfix_branch
```

По завершении работы ветка hotfix объединяется с master и develop:

```
git flow hotfix finish hotfix_branch
```

Семантическое версионирование

Семантический подход в версионированию программного обеспечения.

Краткое описание семантического версионирования

Семантическое версионирование описывается в манифесте семантического версионирования.

Кратко его можно описать следующим образом: Версия задаётся в виде кортежа МАЖОРНАЯ_ВЕРСИЯ.МИНОРНАЯ_ВЕРСИЯ.ПАТЧ. Номер версии следует увеличивать: МАЖОРНУЮ версию, когда сделаны обратно несовместимые изменения API. МИНОРНУЮ версию, когда вы добавляете новую функциональность, не нарушая обратной совместимости. ПАТЧ-версию, когда вы делаете обратно совместимые исправления. Дополнительные обозначения для предрелизных и билд-метаданных возможны как дополнения к МАЖОРНАЯ.МИНОРНАЯ.ПАТЧ

формату.

Программное обеспечение

Для реализации семантического версионирования создано несколько программных продуктов. При этом лучше всего использовать комплексные продукты, которые используют информацию из коммитов системы версионирования. Коммиты должны иметь стандартизованный вид. В семантическое версионирование применяется вместе с общепринятыми коммитами.

Пакет Conventional Changelog Пакет Conventional Changelog является комплексным решением по управлению коммитами и генерации журнала изменений. Содержит набор утилит, которые можно использовать по-отдельности.

Общепринятые коммиты

Использование спецификации Conventional Commits.

Описание

Спецификация Conventional Commits:

Соглашение о том, как нужно писать сообщения commit'ов. Совместимо с SemVer. Даже вернее сказать, сильно связано с семантическим версионированием. Регламентирует структуру и основные типы коммитов.

Структура коммита

():

Или, по-русски:

(): [необязательное тело] [необязательный нижний колонтитул]

Заголовок является обязательным. Любая строка сообщения о фиксации не может быть длиннее 100 символов. Тема (subject) содержит краткое описание изменения. Используйте повелительное наклонение в настоящем времени: «изменить» (“change” not “changed” nor “changes”). Не используйте заглавную первую букву. Не ставьте точку в конце. Тело (body) должно включать мотивацию к изменению и противопоставлять это предыдущему поведению. Как и в теме, используйте повелительное наклонение в настоящем времени. Нижний колонтитул (footer) должен содержать любую информацию о критических изменениях. Сле-

дует использовать для указания внешних ссылок, контекста коммита или другой мета информации. Также содержит ссылку на issue (например, на github), который закрывает эта фиксация. Критические изменения должны начинаться со слова BREAKING CHANGE: с пробела или двух символов новой строки. Затем для этого используется остальная часть сообщения фиксации.

Типы коммитов

Базовые типы коммитов fix: — коммит типа fix исправляет ошибку (bug) в вашем коде (он соответствует PATCH в SemVer). feat: — коммит типа feat добавляет новую функцию (feature) в ваш код (он соответствует MINOR в SemVer). BREAKING CHANGE: — коммит, который содержит текст BREAKING CHANGE: в начале своего не обязательного тела сообщения (body) или в подвале (footer), добавляет изменения, нарушающие обратную совместимость вашего API (он соответствует MAJOR в SemVer). BREAKING CHANGE может быть частью коммита любого типа. revert: — если фиксация отменяет предыдущую фиксацию. Начинается с revert:, за которым следует заголовок отменённой фиксации. В теле должно быть написано: Это отменяет фиксацию (это SHA-хэш отменяемой фиксации). Другое: коммиты с типами, которые отличаются от fix: и feat:, также разрешены. Например, **[commitlint/config-conventional?]** (основанный на The Angular convention) рекомендует: chore:, docs:, style:, refactor:, perf:, test:, и другие.

Соглашения The Angular convention Одно из популярных соглашений о поддержке исходных кодов — конвенция Angular (The Angular convention).

Типы коммитов The Angular convention

Конвенция Angular (The Angular convention) требует следующие типы коммитов: build: — изменения, влияющие на систему сборки или внешние зависимости (примеры областей (scope): gulp, broccoli, npm). ci: — изменения в файлах конфигурации и скриптах CI (примеры областей: Travis, Circle, BrowserStack, SauceLabs). docs: — изменения только в документации. feat: — новая функция. fix: — исправление ошибок. perf: — изменение кода, улучшающее производительность. refactor: — Изменение кода, которое не исправляет ошибку и не добавляет функции (ре-

факторинг кода). `style:` — изменения, не влияющие на смысл кода (пробелы, форматирование, отсутствие точек с запятой и т. д.). `test:` — добавление недостающих тестов или исправление существующих тестов.

Области действия (scope)

Областью действия должно быть имя затронутого пакета `npm` (как его воспринимает человек, читающий журнал изменений, созданный из сообщений фиксации).

Есть несколько исключений из правила «использовать имя пакета»: `packaging` — используется для изменений, которые изменяют структуру пакета, например, изменения общедоступного пути. `changelog` — используется для обновления примечаний к выпуску в `CHANGELOG.md`. отсутствует область действия — полезно для изменений стиля, тестирования и рефакторинга, которые выполняются во всех пакетах (например, `style:` добавить отсутствующие точки с запятой).

Соглашения [**`commitlint/config-conventional?`**]

Соглашение [**`commitlint/config-conventional?`**] входит в пакет `Conventional Changelog`. В целом в этом соглашении придерживаются соглашения `Angular`.

Более подробно о Linux см. в [1-7]

4 Выполнение лабораторной работы

1. Создание репозитория git-extended (рис. 4.1).

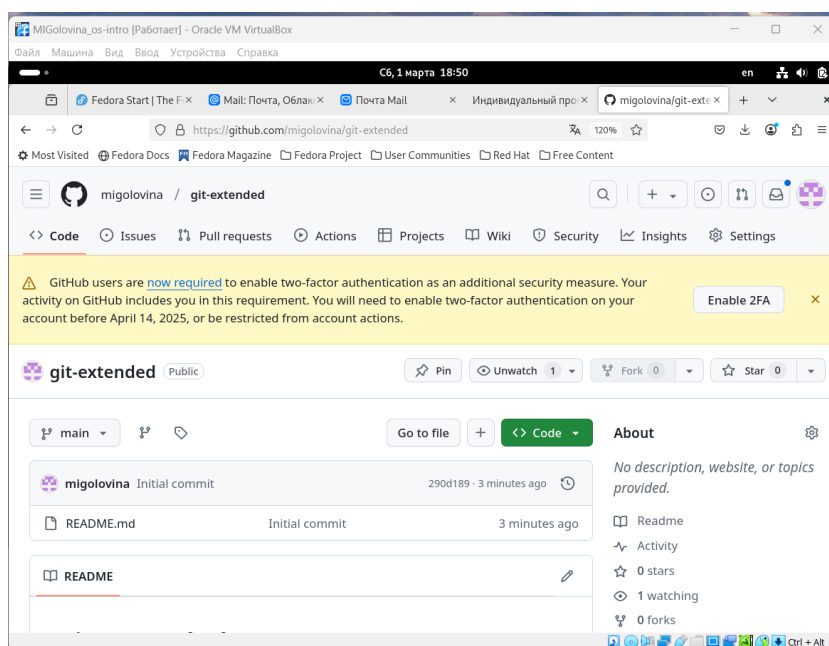
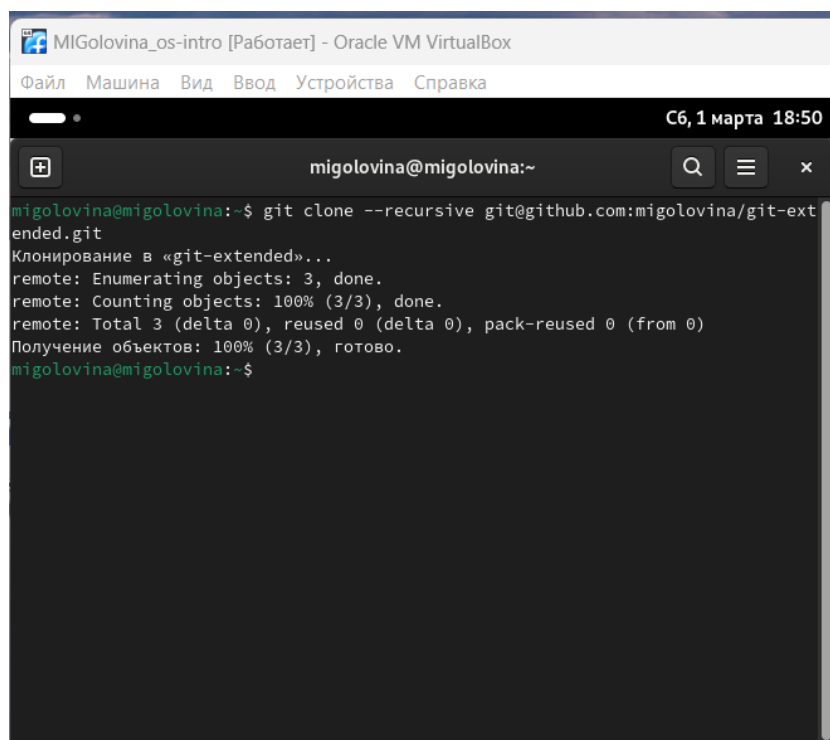


Рис. 4.1: Репозиторий git-extended

2. Клонирование репозитория git-extended (рис. 4.2).

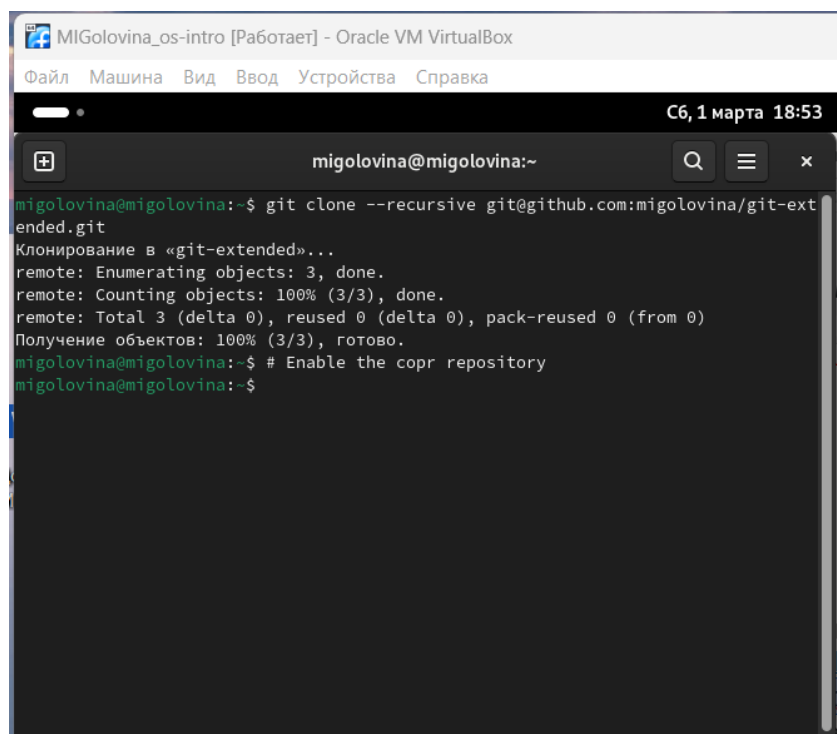


The screenshot shows a terminal window titled "migolovina@migolovina:~" within an Oracle VM VirtualBox environment. The terminal displays the command `git clone --recursive git@github.com:migolovina/git-extended.git` and its output, which includes progress reports for enumerating and counting objects, and a final confirmation that the objects have been successfully cloned. The window's title bar indicates the VM is running and shows the date and time as "С6, 1 марта 18:50".

```
migolovina@migolovina:~$ git clone --recursive git@github.com:migolovina/git-extended.git
Клонирование в «git-extended»...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Получение объектов: 100% (3/3), готово.
migolovina@migolovina:~$
```

Рис. 4.2: Клонирование

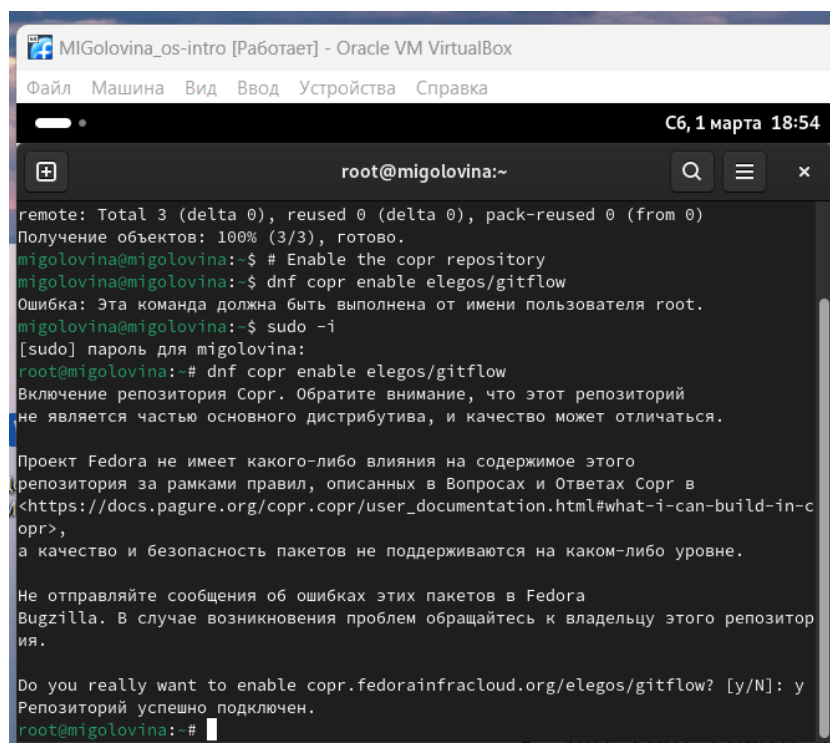
3. Установка из коллекции репозитория Copr. Введение команд: `# Enable the copr repository, dnf copr enable elegos/gitflow` (рис. 4.3-4.4).



The screenshot shows a terminal window titled "migolovina@migolovina:~" with a search icon, a menu icon, and a close icon. The terminal output is as follows:

```
migolovina@migolovina:~$ git clone --recursive git@github.com:migolovina/git-extended.git
Клонирование в «git-extended»...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Получение объектов: 100% (3/3), готово.
migolovina@migolovina:~$ # Enable the copr repository
migolovina@migolovina:~$
```

Рис. 4.3: # Enable the copr repository



```
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Получение объектов: 100% (3/3), готово.
migoLovina@migoLovina:~$ # Enable the copr repository
migoLovina@migoLovina:~$ dnf copr enable elegos/gitflow
Ошибка: Эта команда должна быть выполнена от имени пользователя root.
migoLovina@migoLovina:~$ sudo -i
[sudo] пароль для migoLovina:
root@migoLovina:~$ dnf copr enable elegos/gitflow
Включение репозитория Copr. Обратите внимание, что этот репозиторий
не является частью основного дистрибутива, и качество может отличаться.

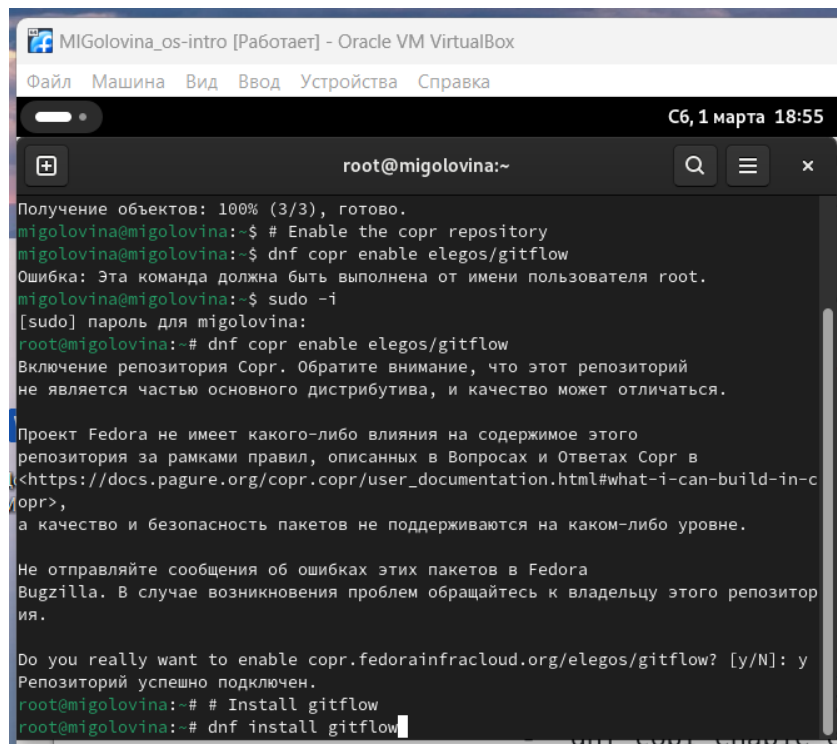
Проект Fedora не имеет какого-либо влияния на содержимое этого
репозитория за рамками правил, описанных в Вопросах и Ответах Copr в
<https://docs.pagure.org/copr.copr/user\_documentation.html#what-i-can-build-in-copr>,
а качество и безопасность пакетов не поддерживаются на каком-либо уровне.

Не отправляйте сообщения об ошибках этих пакетов в Fedora
Bugzilla. В случае возникновения проблем обращайтесь к владельцу этого репозитория.

Do you really want to enable copr.fedorainfracloud.org/elegos/gitflow? [y/N]: y
Репозиторий успешно подключен.
root@migoLovina:~$
```

Рис. 4.4: dnf copr enable elegos/gitflow

4. Установка из коллекции репозиторияев Copr. Введение команд: # Install gitflow, dnf install gitflow (рис. 4.5-4.6).



```
Получение объектов: 100% (3/3), готово.
migolovina@migolovina:~$ # Enable the copr repository
migolovina@migolovina:~$ dnf copr enable elegos/gitflow
Ошибка: Эта команда должна быть выполнена от имени пользователя root.
migolovina@migolovina:~$ sudo -i
[sudo] пароль для migolovina:
root@migolovina:~# dnf copr enable elegos/gitflow
Включение репозитория Copr. Обратите внимание, что этот репозиторий
не является частью основного дистрибутива, и качество может отличаться.

Проект Fedora не имеет какого-либо влияния на содержимое этого
репозитория за рамками правил, описанных в Вопросах и Ответах Copr в
<https://docs.pagure.org/copr.copr/user\_documentation.html#what-i-can-build-in-copr>,
а качество и безопасность пакетов не поддерживаются на каком-либо уровне.

Не отправляйте сообщения об ошибках этих пакетов в Fedora
Bugzilla. В случае возникновения проблем обращайтесь к владельцу этого репозитория.

Do you really want to enable copr.fedorainfracloud.org/elegos/gitflow? [y/N]: y
Репозиторий успешно подключен.
root@migolovina:~# # Install gitflow
root@migolovina:~# dnf install gitflow
```

Рис. 4.5: # Install gitflow

```

MIGolovina_os-intro [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
C6, 1 марта 18:55

root@migolovina:~
а качество и безопасность пакетов не поддерживаются на каком-либо уровне.

Не отправляйте сообщения об ошибках этих пакетов в Fedora
Bugzilla. В случае возникновения проблем обращайтесь к владельцу этого репозитория.

Do you really want to enable copr.fedorainfracloud.org/elegos/gitflow? [y/N]: y
Репозиторий успешно подключен.
root@migolovina:~ # Install gitflow
root@migolovina:~ # dnf install gitflow
Copr repo for gitflow owned by elegos          3.8 kB/s | 1.5 kB    00:00
Copr repo for PyCharm owned by phracek         9.4 kB/s | 2.1 kB    00:00
Fedora 40 - x86_64 - Updates                   28 kB/s | 24 kB     00:00
Fedora 40 - x86_64 - Updates                   4.4 MB/s | 5.1 MB    00:01
google-chrome                                17 kB/s | 1.3 kB    00:00
google-chrome                                10 kB/s | 1.9 kB    00:00
RPM Fusion for Fedora 40 - Nonfree - NVIDIA Dri 14 kB/s | 8.9 kB    00:00
RPM Fusion for Fedora 40 - Nonfree - NVIDIA Dri 8.4 kB/s | 5.2 kB    00:00
RPM Fusion for Fedora 40 - Nonfree - Steam      52 kB/s | 8.9 kB    00:00
Пакет gitflow-1.12.3-1.fc34.x86_64 уже установлен.
Зависимости разрешены.
Нет действий для выполнения.
Выполнено!
root@migolovina:~ #
```

Рис. 4.6: dnf install gitflow

5. Установка Node.js (рис. 4.7-4.8).

The screenshot shows a terminal window titled "root@migolovina:~" with a search icon, a menu icon, and a close icon. The terminal output is as follows:

```
Репозиторий успешно подключен.
root@migolovina:~# # Install gitflow
root@migolovina:~# dnf install gitflow
Copr repo for gitflow owned by elegos          3.8 kB/s | 1.5 kB      00:00
Copr repo for PyCharm owned by phracek         9.4 kB/s | 2.1 kB      00:00
Fedora 40 - x86_64 - Updates                   28 kB/s | 24 kB        00:00
Fedora 40 - x86_64 - Updates                   4.4 MB/s | 5.1 MB       00:01
google-chrome                                17 kB/s | 1.3 kB       00:00
google-chrome                                10 kB/s | 1.9 kB       00:00
RPM Fusion for Fedora 40 - Nonfree - NVIDIA Dri 14 kB/s | 8.9 kB       00:00
RPM Fusion for Fedora 40 - Nonfree - NVIDIA Dri 8.4 kB/s | 5.2 kB       00:00
RPM Fusion for Fedora 40 - Nonfree - Steam     52 kB/s | 8.9 kB       00:00
Пакет gitflow-1.12.3-1.fc34.x86_64 уже установлен.
Зависимости разрешены.
Нет действий для выполнения.
Выполнено!
root@migolovina:~# dnf install nodejs
Последняя проверка окончания срока действия метаданных: 0:00:40 назад, Сб 01 мар 2025 18:55:24.
Пакет nodejs-1:20.18.2-2.fc40.x86_64 уже установлен.
Зависимости разрешены.
Нет действий для выполнения.
Выполнено!
root@migolovina:~#
```

Рис. 4.7: dnf install nodejs

The screenshot shows a terminal window titled "root@migolovina:~" within an Oracle VM VirtualBox environment. The window displays the output of several dnf commands. At the top, there is a table showing the progress of installing packages: google-chrome, RPM Fusion for Fedora 40 - Nonfree - NVIDIA Dri, and RPM Fusion for Fedora 40 - Nonfree - Steam. Below this, it states that the gitflow-1.12.3-1.fc34.x86_64 package is already installed and its dependencies are resolved. Then, the command "dnf install nodejs" is executed, showing that nodejs-1:20.18.2-2.fc40.x86_64 is already installed and its dependencies are resolved. Finally, the command "dnf install pnpm" is executed, showing that pnpm-9.0.6-1.fc40.noarch is already installed and its dependencies are resolved. The terminal window has a dark background and a light-colored text.

```
MIGolovina_os-intro [Работает] - Oracle VM VirtualBox
Файл Машина Вид Ввод Устройства Справка
C6, 1 марта 18:56
root@migolovina:~
google-chrome 17 kB/s | 1.3 kB 00:00
google-chrome 10 kB/s | 1.9 kB 00:00
RPM Fusion for Fedora 40 - Nonfree - NVIDIA Dri 14 kB/s | 8.9 kB 00:00
RPM Fusion for Fedora 40 - Nonfree - NVIDIA Dri 8.4 kB/s | 5.2 kB 00:00
RPM Fusion for Fedora 40 - Nonfree - Steam 52 kB/s | 8.9 kB 00:00
Пакет gitflow-1.12.3-1.fc34.x86_64 уже установлен.
Зависимости разрешены.
Нет действий для выполнения.
Выполнено!
root@migolovina:~# dnf install nodejs
Последняя проверка окончания срока действия метаданных: 0:00:40 назад, C6 01 мар
2025 18:55:24.
Пакет nodejs-1:20.18.2-2.fc40.x86_64 уже установлен.
Зависимости разрешены.
Нет действий для выполнения.
Выполнено!
root@migolovina:~# dnf install pnpm
Последняя проверка окончания срока действия метаданных: 0:01:05 назад, C6 01 мар
2025 18:55:24.
Пакет pnpm-9.0.6-1.fc40.noarch уже установлен.
Зависимости разрешены.
Нет действий для выполнения.
Выполнено!
root@migolovina:~#
```

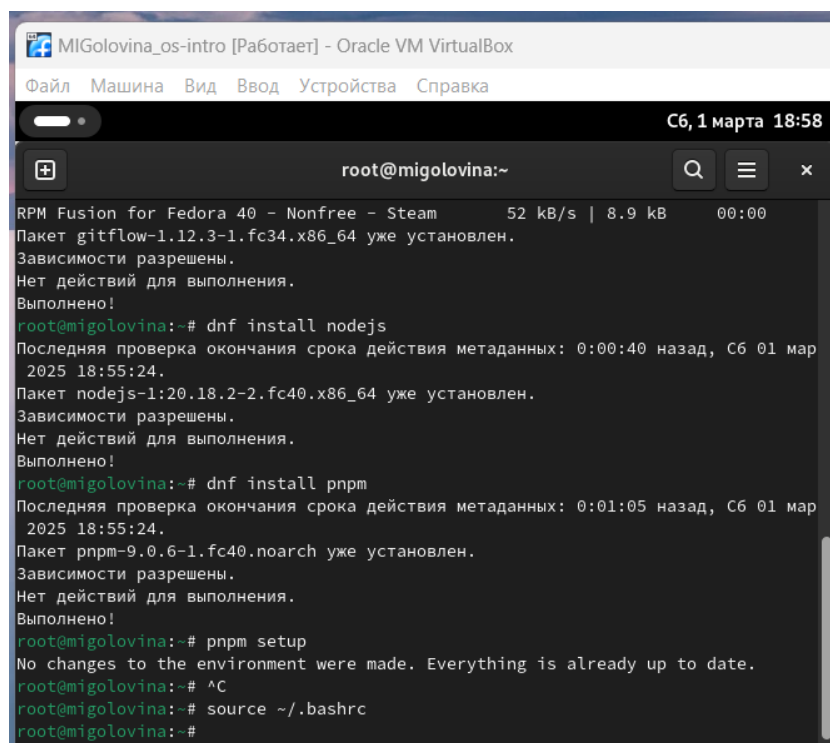
Рис. 4.8: dnf install pnpm

6. Настройка Node.js (рис. 4.9-4.10).

The screenshot shows a terminal window titled "MIgolovina_os-intro [Работает] - Oracle VM VirtualBox". The terminal prompt is "root@migolovina:~". The output shows the installation of nodejs and pnpm using dnf, followed by the pnpm setup command which reports no changes to the environment.

```
root@migolovina:~  
RPM Fusion for Fedora 40 - Nonfree - NVIDIA Dri 8.4 kB/s | 5.2 kB 00:00  
RPM Fusion for Fedora 40 - Nonfree - Steam 52 kB/s | 8.9 kB 00:00  
Пакет gitflow-1.12.3-1.fc34.x86_64 уже установлен.  
Зависимости разрешены.  
Нет действий для выполнения.  
Выполнено!  
root@migolovina:~# dnf install nodejs  
Последняя проверка окончания срока действия метаданных: 0:00:40 назад, Сб 01 мар  
2025 18:55:24.  
Пакет nodejs-1:20.18.2-2.fc40.x86_64 уже установлен.  
Зависимости разрешены.  
Нет действий для выполнения.  
Выполнено!  
root@migolovina:~# dnf install pnpm  
Последняя проверка окончания срока действия метаданных: 0:01:05 назад, Сб 01 мар  
2025 18:55:24.  
Пакет pnpm-9.0.6-1.fc40.noarch уже установлен.  
Зависимости разрешены.  
Нет действий для выполнения.  
Выполнено!  
root@migolovina:~# pnpm setup  
No changes to the environment were made. Everything is already up to date.  
root@migolovina:~# ^C  
root@migolovina:~#
```

Рис. 4.9: pnpm setup

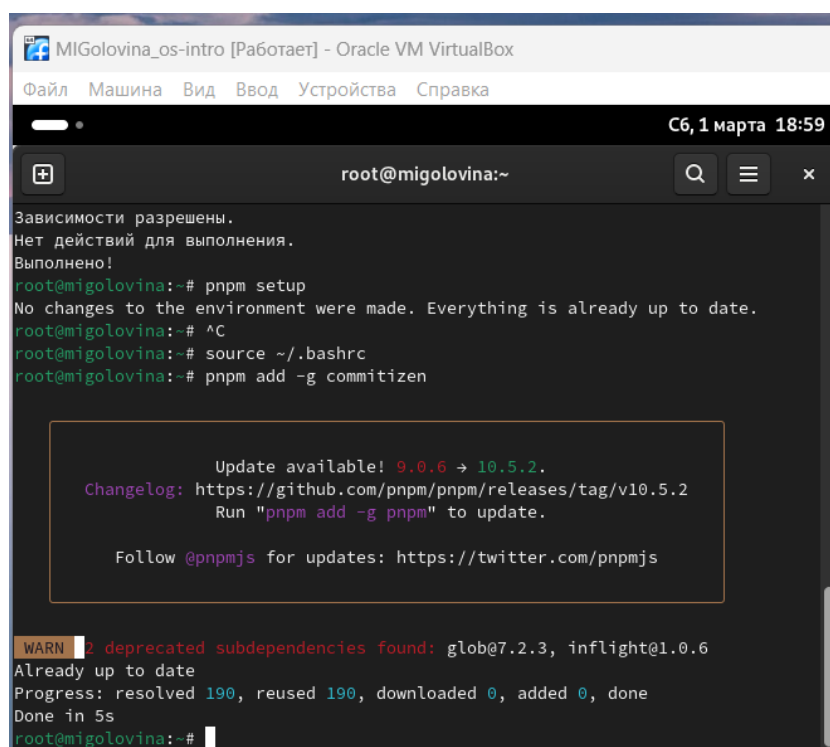


The screenshot shows a terminal window titled "root@migolovina:~" within an Oracle VM VirtualBox environment. The terminal output shows the installation of Node.js and npm using dnf, followed by the execution of the pnpm setup command and the source command to reload the environment. The terminal text is as follows:

```
RPM Fusion for Fedora 40 - Nonfree - Steam 52 kB/s | 8.9 kB 00:00
Пакет gitflow-1.12.3-1.fc34.x86_64 уже установлен.
Зависимости разрешены.
Нет действий для выполнения.
Выполнено!
root@migolovina:~# dnf install nodejs
Последняя проверка окончания срока действия метаданных: 0:00:40 назад, Сб 01 мар 2025 18:55:24.
Пакет nodejs-1:20.18.2-2.fc40.x86_64 уже установлен.
Зависимости разрешены.
Нет действий для выполнения.
Выполнено!
root@migolovina:~# dnf install pnpm
Последняя проверка окончания срока действия метаданных: 0:01:05 назад, Сб 01 мар 2025 18:55:24.
Пакет pnpm-9.0.6-1.fc40.noarch уже установлен.
Зависимости разрешены.
Нет действий для выполнения.
Выполнено!
root@migolovina:~# pnpm setup
No changes to the environment were made. Everything is already up to date.
root@migolovina:~# ^C
root@migolovina:~# source ~/.bashrc
root@migolovina:~#
```

Рис. 4.10: source ~/.bashrc

7. Общепринятые коммиты (рис. 4.11-4.12).



The screenshot shows a terminal window titled "root@migolovina:~" within an Oracle VM VirtualBox environment. The terminal displays the following sequence of commands and their outputs:

```
Зависимости разрешены.  
Нет действий для выполнения.  
Выполнено!  
root@migolovina:~# pnpm setup  
No changes to the environment were made. Everything is already up to date.  
root@migolovina:~# ^C  
root@migolovina:~# source ~/.bashrc  
root@migolovina:~# pnpm add -g commitizen
```

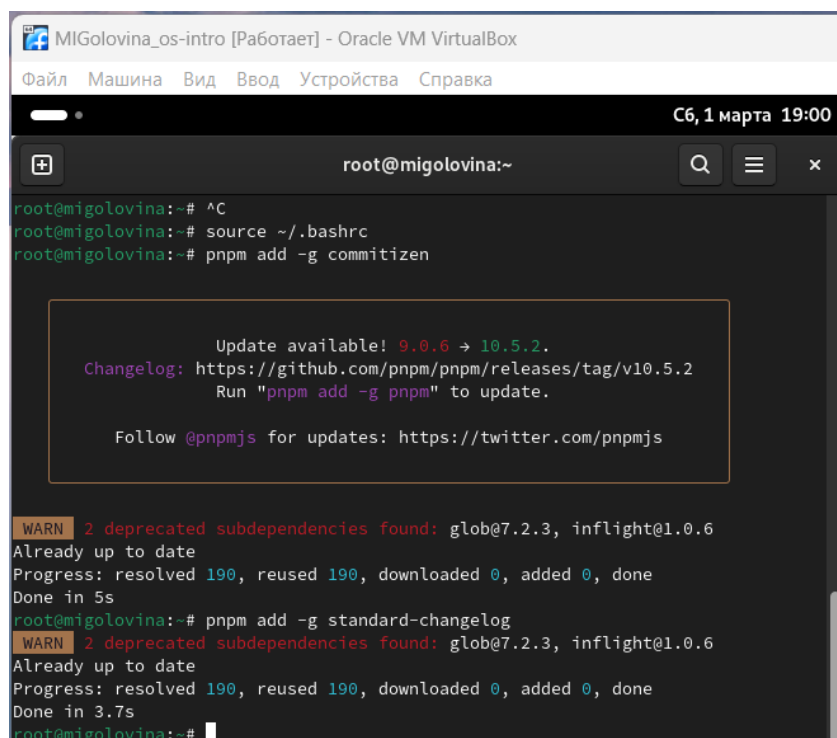
A message box highlights an update for pnpm:

```
Update available! 9.0.6 → 10.5.2.  
Changelog: https://github.com/pnpm/pnpm/releases/tag/v10.5.2  
Run "pnpm add -g pnpm" to update.  
  
Follow @pnpmjs for updates: https://twitter.com/pnpmjs
```

Below the message box, the terminal shows a warning and completion status:

```
WARN deprecated subdependencies found: glob@7.2.3, inflight@1.0.6  
Already up to date  
Progress: resolved 190, reused 190, downloaded 0, added 0, done  
Done in 5s  
root@migolovina:~#
```

Рис. 4.11: pnpm add -g commitizen



The screenshot shows a terminal window titled "MiGolovina_os-intro [Работает] - Oracle VM VirtualBox". The terminal prompt is "root@migolovina:~". The user enters the following commands and receives the following output:

```
root@migolovina:~# ^C
root@migolovina:~# source ~/.bashrc
root@migolovina:~# pnpm add -g commitizen
```

An update notification box appears with the following text:

```
Update available! 9.0.6 → 10.5.2.
Changelog: https://github.com/pnpm/pnpm/releases/tag/v10.5.2
Run "pnpm add -g pnpm" to update.
Follow @pnpmjs for updates: https://twitter.com/pnpmjs
```

The terminal then shows the output of the command:

```
WARN 2 deprecated subdependencies found: glob@7.2.3, inflight@1.0.6
Already up to date
Progress: resolved 190, reused 190, downloaded 0, added 0, done
Done in 5s
```

The user then enters the command:

```
root@migolovina:~# pnpm add -g standard-changelog
```

The output is:

```
WARN 2 deprecated subdependencies found: glob@7.2.3, inflight@1.0.6
Already up to date
Progress: resolved 190, reused 190, downloaded 0, added 0, done
Done in 3.7s
root@migolovina:~#
```

Рис. 4.12: pnpm add -g standard-changelog

8. Делаем первый коммит (рис. 4.13-4.14).

The screenshot shows a terminal window titled "migolovina@migolovina:~/git-extended". The user has entered the command `git commit -m "first commit"`. The terminal output indicates that the current branch is `main` and that it corresponds to `origin/main`. It also states that there are no changes to commit because the working directory is clean. The terminal window is part of a larger application window titled "MIGolovina_os-intro [Работает] - Oracle VM VirtualBox".

```
migolovina@migolovina:~/git-extended$ git commit -m "first commit"
Текущая ветка: main
Эта ветка соответствует «origin/main».

нечего коммитить, нет изменений в рабочем каталоге
migolovina@migolovina:~/git-extended$
```

Рис. 4.13: git commit -m “first commit”

The screenshot shows a terminal window titled "migolovina@migolovina:~/git-extended". The user has entered the command `git commit -m "first commit"`. The terminal output indicates that the current branch is `main` and that it corresponds to `origin/main`. It also states that there are no changes to commit because the working directory is clean. The user then enters the command `git remote add origin git@github.com:migolovina/git-extended.git`. The terminal output shows that the remote has been added successfully. The terminal window is part of a larger application window titled "MIGolovina_os-intro [Работает] - Oracle VM VirtualBox".

```
migolovina@migolovina:~/git-extended$ git commit -m "first commit"
Текущая ветка: main
Эта ветка соответствует «origin/main».

нечего коммитить, нет изменений в рабочем каталоге
migolovina@migolovina:~/git-extended$ git commit -m "first commit"
Текущая ветка: main
Эта ветка соответствует «origin/main».

Неотслеживаемые файлы:
(используйте «git add <файл>...», чтобы добавить в то, что будет включено в коммит)
text.txt

индекс пуст, но есть неотслеживаемые файлы
(используйте «git add», чтобы проиндексировать их)
migolovina@migolovina:~/git-extended$ git remote add origin git@github.com:migolovina/git-extended.git
```

Рис. 4.14: git remote add origin git@github.com:migolovina/git-extended.git

9. Выкладываем на github (рис. 4.15).

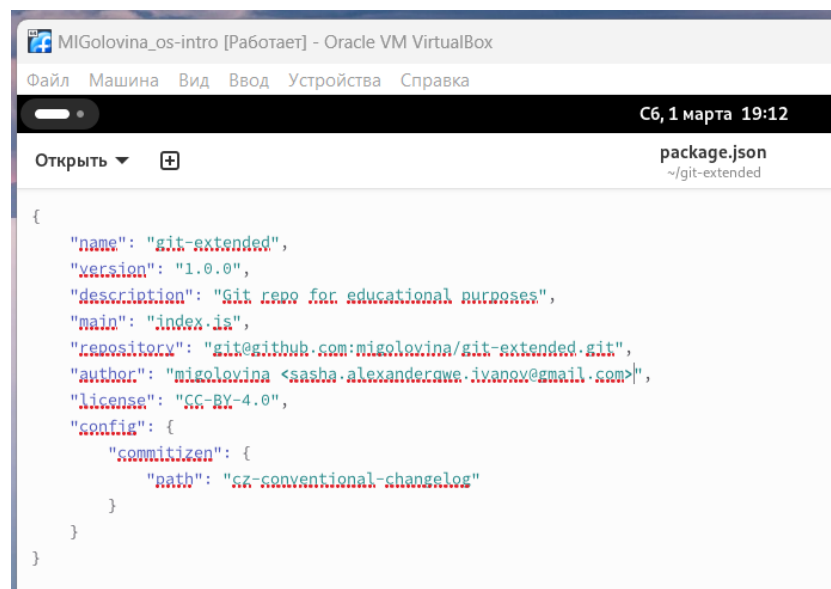
```
migolovina@migolovina:~/git-extended$ git commit -m "first commit"
Текущая ветка: main
Неотслеживаемые файлы:
(используйте «git add <файл>...», чтобы добавить в то, что будет включено в коммит)

    text.txt

индекс пуст, но есть неотслеживаемые файлы
(используйте «git add», чтобы проиндексировать их)
migolovina@migolovina:~/git-extended$ git add .
migolovina@migolovina:~/git-extended$ git push -u origin main
branch 'main' set up to track 'origin/main'.
Everything up-to-date
migolovina@migolovina:~/git-extended$
```

Рис. 4.15: git push -u origin main

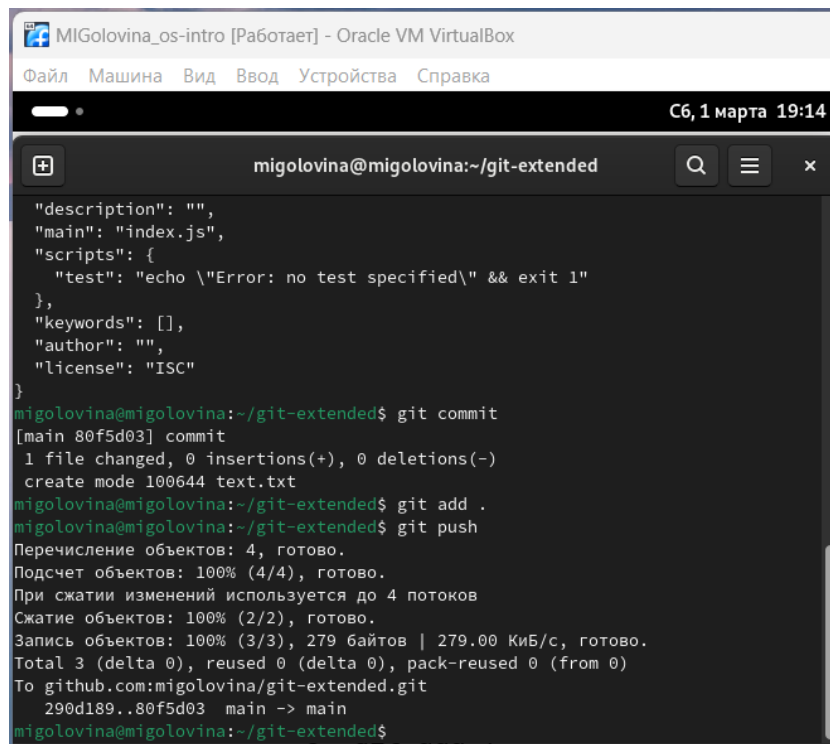
10. Конфигурация общепринятых коммитов. (рис. 4.16-4.17).



The screenshot shows a code editor window titled "MIGolovina_os-intro [Работает] - Oracle VM VirtualBox". The editor displays the content of a file named "package.json" located at "~/git-extended". The JSON content is as follows:

```
{
  "name": "git-extended",
  "version": "1.0.0",
  "description": "Git repo for educational purposes",
  "main": "index.js",
  "repository": "git@github.com:migolovina/git-extended.git",
  "author": "migolovina <sasha.alexanderawa.ivanov@gmail.com>",
  "license": "CC-BY-4.0",
  "config": {
    "commitizen": {
      "path": "cz-conventional-changelog"
    }
  }
}
```

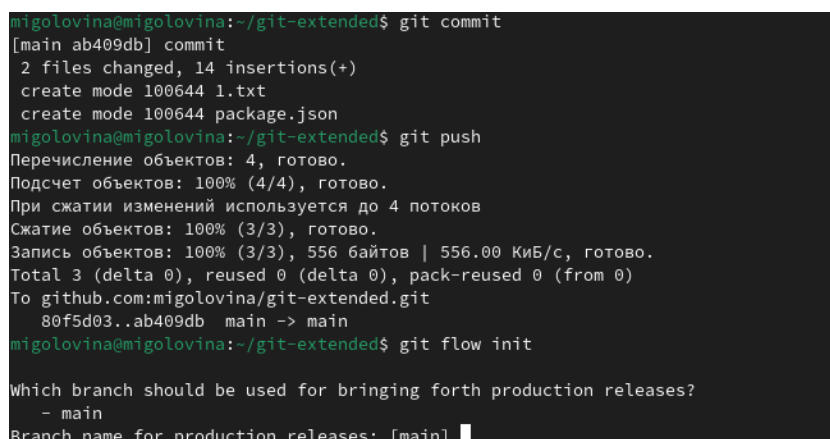
Рис. 4.16: package.json



The screenshot shows a terminal window titled "migolovina@migolovina:~/git-extended". The terminal displays the following commands and output:

```
"description": "",
"main": "index.js",
"scripts": {
  "test": "echo \"Error: no test specified\" && exit 1"
},
"keywords": [],
"author": "",
"license": "ISC"
}
migolovina@migolovina:~/git-extended$ git commit
[main 80f5d03] commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 text.txt
migolovina@migolovina:~/git-extended$ git add .
migolovina@migolovina:~/git-extended$ git push
Перечисление объектов: 4, готово.
Подсчет объектов: 100% (4/4), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (2/2), готово.
Запись объектов: 100% (3/3), 279 байтов | 279.00 КиБ/с, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:migolovina/git-extended.git
290d189..80f5d03 main -> main
migolovina@migolovina:~/git-extended$
```

Рис. 4.17: Добавляем новые файлы, выполняем коммит и отправляем на github



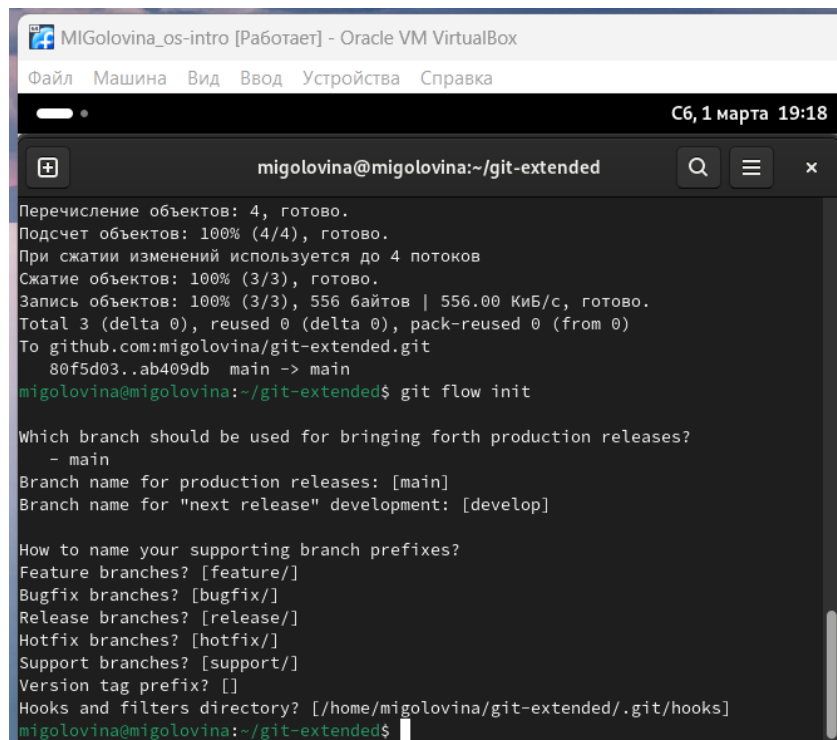
The screenshot shows a terminal window titled "migolovina@migolovina:~/git-extended". The terminal displays the following commands and output:

```
migolovina@migolovina:~/git-extended$ git commit
[main ab409db] commit
2 files changed, 14 insertions(+)
create mode 100644 l.txt
create mode 100644 package.json
migolovina@migolovina:~/git-extended$ git push
Перечисление объектов: 4, готово.
Подсчет объектов: 100% (4/4), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 556 байтов | 556.00 КиБ/с, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:migolovina/git-extended.git
80f5d03..ab409db main -> main
migolovina@migolovina:~/git-extended$ git flow init

Which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main]
```

Рис. 4.18: Отправляем на github

11. Конфигурация git-flow. (рис. 4.19-4.31).



```
MIGolovina_os-intro [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
C6, 1 марта 19:18
migolovina@migolovina:~/git-extended
Перечисление объектов: 4, готово.
Подсчет объектов: 100% (4/4), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 556 байтов | 556.00 КиБ/с, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:migolovina/git-extended.git
80f5d03..ab409db  main -> main
migolovina@migolovina:~/git-extended$ git flow init

Which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [/home/migolovina/git-extended/.git/hooks]
migolovina@migolovina:~/git-extended$
```

Рис. 4.19: Инициализируем git-flow

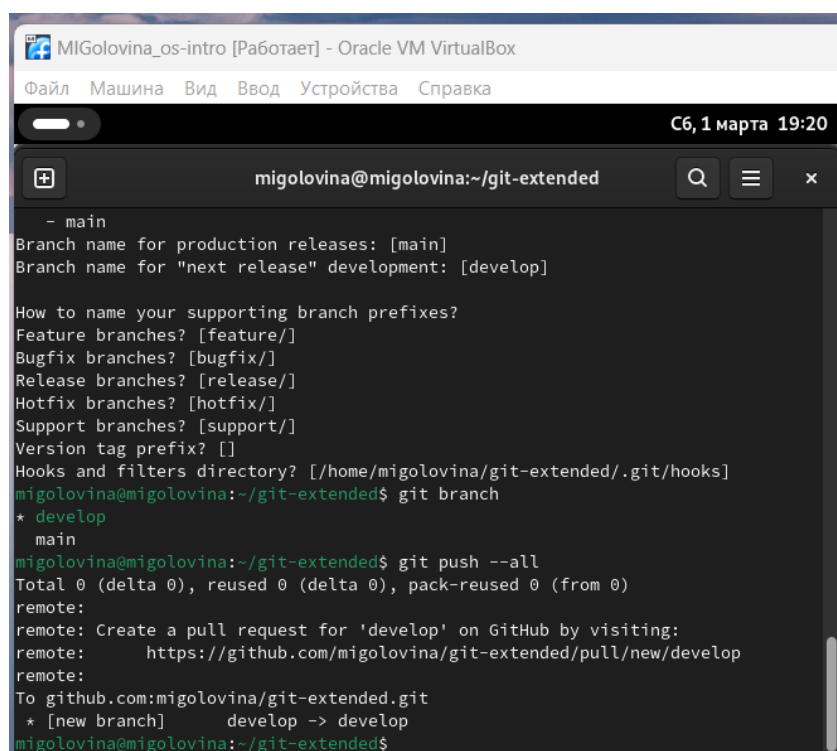
The screenshot shows a terminal window titled "migolovina@migolovina:~/git-extended". The terminal output shows the completion of a Git push to GitHub, followed by the execution of "git flow init". This command prompts the user to configure Git Flow, including selecting the main branch for production releases, the development branch for the next release, and various branch prefixes. Finally, the "git branch" command is run, showing the current branches: "develop" (marked with an asterisk) and "main".

```
MIGolovina_os-intro [Работает] - Oracle VM VirtualBox
Файл Машина Вид Ввод Устройства Справка
С6, 1 марта 19:19
migolovina@migolovina:~/git-extended
Сжатие объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 556 байтов | 556.00 КиБ/с, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:migolovina/git-extended.git
80f5d03..ab409db main -> main
migolovina@migolovina:~/git-extended$ git flow init

Which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [/home/migolovina/git-extended/.git/hooks]
migolovina@migolovina:~/git-extended$ git branch
* develop
main
migolovina@migolovina:~/git-extended$
```

Рис. 4.20: Проверка



The screenshot shows a terminal window titled "migolovina@migolovina:~/git-extended". The terminal output displays the following sequence of commands and responses:

```
- main
Branch name for production releases: [main]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [/home/migolovina/git-extended/.git/hooks]
migolovina@migolovina:~/git-extended$ git branch
* develop
  main
migolovina@migolovina:~/git-extended$ git push --all
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:   https://github.com/migolovina/git-extended/pull/new/develop
remote:
To github.com:migolovina/git-extended.git
 * [new branch]      develop -> develop
migolovina@migolovina:~/git-extended$
```

Рис. 4.21: Загрузка всего репозитория в хранилище

The screenshot shows a terminal window titled "migolovina@migolovina:~/git-extended". The terminal displays the following commands and output:

```
How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [/home/migolovina/git-extended/.git/hooks]
migolovina@migolovina:~/git-extended$ git branch
* develop
  main
migolovina@migolovina:~/git-extended$ git push --all
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:   https://github.com/migolovina/git-extended/pull/new/develop
remote:
To github.com:migolovina/git-extended.git
 * [new branch]      develop -> develop
migolovina@migolovina:~/git-extended$ git branch --set-upstream-to=origin/develop develop
branch 'develop' set up to track 'origin/develop'.
migolovina@migolovina:~/git-extended$
```

Рис. 4.22: Установка внешней ветки как вышестоящую для этой ветки

The screenshot shows a terminal window titled "migolovina@migolovina:~/git-extended" within an Oracle VM VirtualBox environment. The terminal displays the output of several Git commands. It starts with a remote message about creating a pull request. Then, the user runs `git branch --set-upstream-to=origin/develop develop`, which creates a new branch 'develop' and sets it up to track 'origin/develop'. Next, the user runs `git flow release start 1.0.0`, which creates a new branch 'release/1.0.0' based on 'develop' and switches to it. The terminal then shows a summary of actions and follow-up instructions for completing the release.

```
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:   https://github.com/migolovina/git-extended/pull/new/develop
remote:
To github.com:migolovina/git-extended.git
* [new branch]      develop -> develop
migolovina@migolovina:~/git-extended$ git branch --set-upstream-to=origin/develop develop
branch 'develop' set up to track 'origin/develop'.
migolovina@migolovina:~/git-extended$ git flow release start 1.0.0
Переключились на новую ветку «release/1.0.0»

Summary of actions:
- A new branch 'release/1.0.0' was created, based on 'develop'
- You are now on branch 'release/1.0.0'

Follow-up actions:
- Bump the version number now!
- Start committing last-minute fixes in preparing your release
- When done, run:

    git flow release finish '1.0.0'

migolovina@migolovina:~/git-extended$
```

Рис. 4.23: Создание релиза с версией 1.0.0

```
MIGolovina_os-intro [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка

C6, 1 марта 19:24

migolovina@migolovina:~/git-extended

Summary of actions:
- A new branch 'release/1.0.0' was created, based on 'develop'
- You are now on branch 'release/1.0.0'

Follow-up actions:
- Bump the version number now!
- Start committing last-minute fixes in preparing your release
- When done, run:

    git flow release finish '1.0.0'

migolovina@migolovina:~/git-extended$ npx standard-changelog --first-release
Need to install the following packages:
standard-changelog@6.0.0
Ok to proceed? (y) y

✓ created CHANGELOG.md
✓ output changes to CHANGELOG.md
npm notice
npm notice New major version of npm available! 10.8.2 -> 11.1.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v11.1.0
npm notice To update run: npm install -g npm@11.1.0
npm notice
migolovina@migolovina:~/git-extended$
```

Рис. 4.24: Создании журнала изменений

The screenshot shows a terminal window titled 'migolovina@migolovina:~/git-extended'. The window contains the following text:

```
- A new branch 'release/1.0.0' was created, based on 'develop'
- You are now on branch 'release/1.0.0'

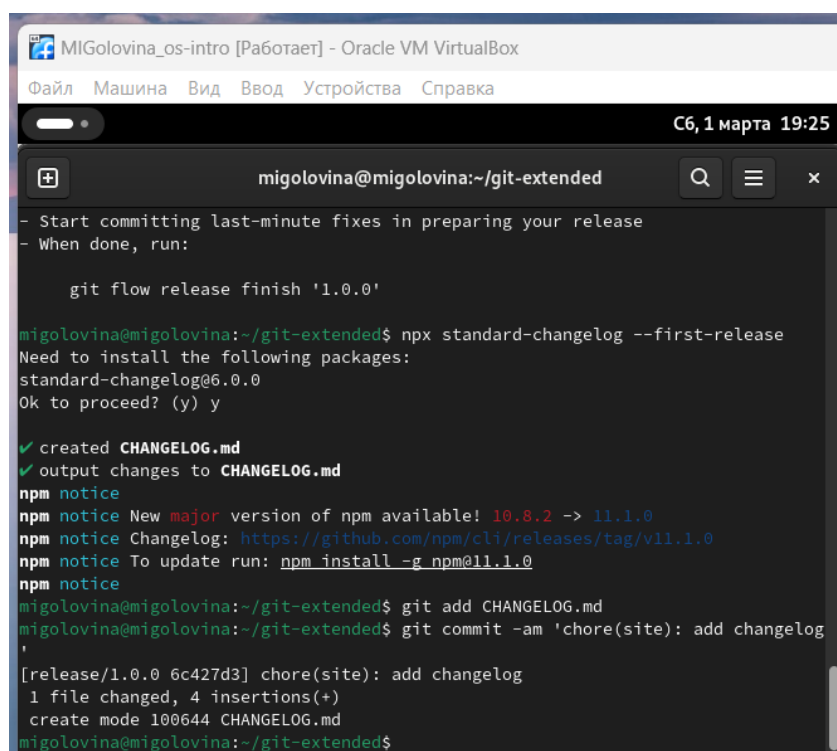
Follow-up actions:
- Bump the version number now!
- Start committing last-minute fixes in preparing your release
- When done, run:

    git flow release finish '1.0.0'

migolovina@migolovina:~/git-extended$ npx standard-changelog --first-release
Need to install the following packages:
standard-changelog@6.0.0
Ok to proceed? (y) y

✓ created CHANGELOG.md
✓ output changes to CHANGELOG.md
npm notice
npm notice New major version of npm available! 10.8.2 -> 11.1.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v11.1.0
npm notice To update run: npm install -g npm@11.1.0
npm notice
migolovina@migolovina:~/git-extended$ git add CHANGELOG.md
migolovina@migolovina:~/git-extended$
```

Рис. 4.25: Добавление журнала изменений в индекс



The screenshot shows a terminal window titled "MIGolovina_os-intro [Работает] - Oracle VM VirtualBox". The terminal is running a series of commands to generate and commit a changelog. The commands and their outputs are as follows:

```
migolovina@migolovina:~/git-extended$ npx standard-changelog --first-release
Need to install the following packages:
standard-changelog@6.0.0
Ok to proceed? (y) y

✓ created CHANGELOG.md
✓ output changes to CHANGELOG.md
npm notice
npm notice New major version of npm available! 10.8.2 -> 11.1.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v11.1.0
npm notice To update run: npm install -g npm@11.1.0
npm notice
migolovina@migolovina:~/git-extended$ git add CHANGELOG.md
migolovina@migolovina:~/git-extended$ git commit -am 'chore(site): add changelog'

[release/1.0.0 6c427d3] chore(site): add changelog
1 file changed, 4 insertions(+)
create mode 100644 CHANGELOG.md
migolovina@migolovina:~/git-extended$
```

Рис. 4.26: Добавление журнала изменений в индекс

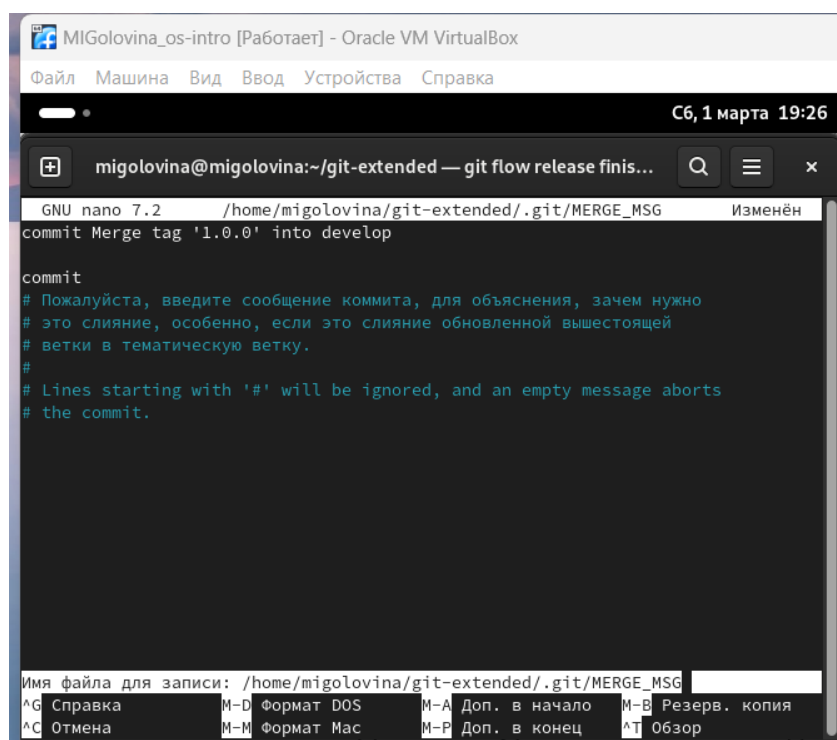
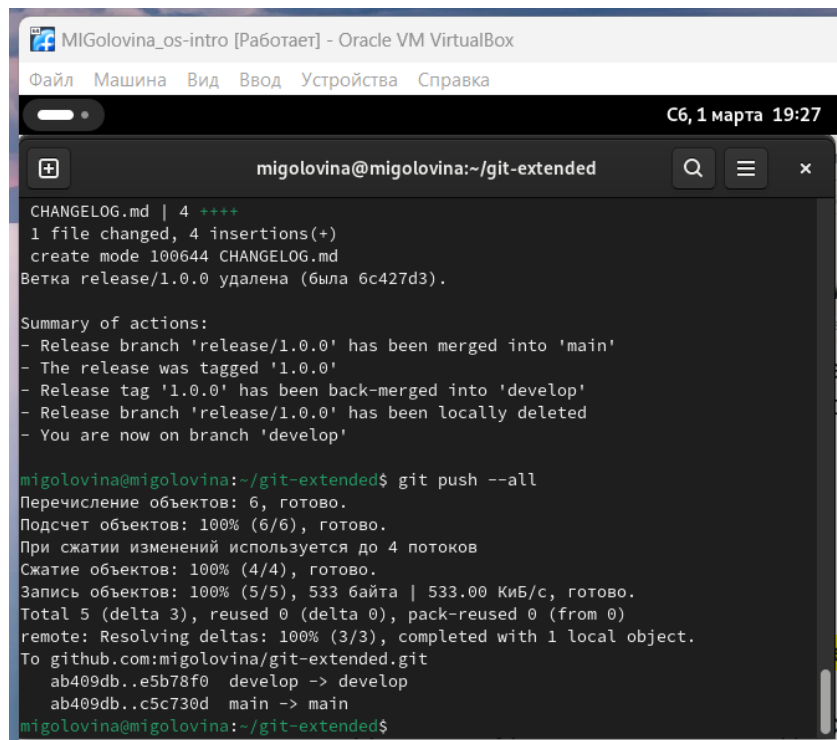


Рис. 4.27: Залитие релизной ветки в основную ветку



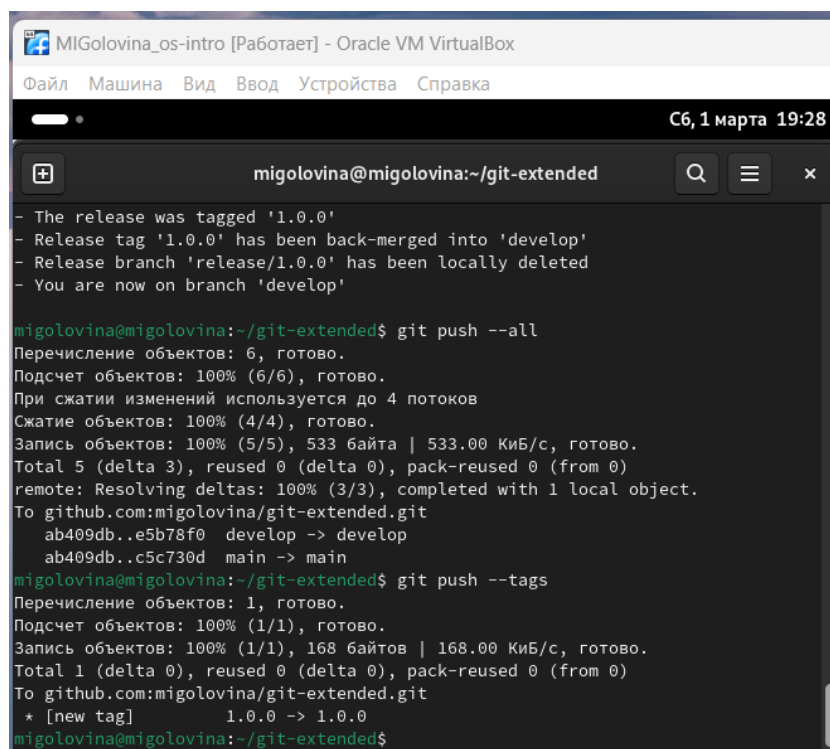
The screenshot shows a terminal window titled "migolovina@migolovina:~/git-extended". The terminal output displays the result of a git push --all command. It shows that a file named CHANGELOG.md was changed with 4 insertions. A summary of actions follows, indicating that the release branch 'release/1.0.0' was merged into 'main', the release was tagged '1.0.0', the tag was back-merged into 'develop', and the release branch was locally deleted. The push command then executes, showing progress for object enumeration, counting, compression, and writing. The final output shows the commit being pushed to the 'develop' branch on the remote repository 'github.com:migolovina/git-extended.git'.

```
CHANGELOG.md | 4 ++++
1 file changed, 4 insertions(+)
create mode 100644 CHANGELOG.md
Ветка release/1.0.0 удалена (была 6c427d3).

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

migolovina@migolovina:~/git-extended$ git push --all
Перечисление объектов: 6, готово.
Подсчет объектов: 100% (6/6), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (4/4), готово.
Запись объектов: 100% (5/5), 533 байта | 533.00 КиБ/с, готово.
Total 5 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 1 local object.
To github.com:migolovina/git-extended.git
   ab409db..e5b78f0  develop -> develop
   ab409db..c5c730d  main -> main
migolovina@migolovina:~/git-extended$
```

Рис. 4.28: Отправление данных на github



The screenshot shows a terminal window titled "migolovina@migolovina:~/git-extended". The window displays the output of two git push commands. The first command is "git push --all", which pushes all local branches to the remote repository. The output shows that 6 objects are enumerated, 100% of the objects are counted, and 5 objects are written to the remote repository. The second command is "git push --tags", which pushes all local tags to the remote repository. The output shows that 1 object is enumerated, 100% of the objects are counted, and 1 tag is written to the remote repository.

```
migolovina@migolovina:~/git-extended$ git push --all
Перечисление объектов: 6, готово.
Подсчет объектов: 100% (6/6), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (4/4), готово.
Запись объектов: 100% (5/5), 533 байта | 533.00 КиБ/с, готово.
Total 5 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 1 local object.
To github.com:migolovina/git-extended.git
   ab409db..e5b78f0  develop -> develop
   ab409db..c5c730d  main -> main
migolovina@migolovina:~/git-extended$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 168 байтов | 168.00 КиБ/с, готово.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:migolovina/git-extended.git
 * [new tag]         1.0.0 -> 1.0.0
migolovina@migolovina:~/git-extended$
```

Рис. 4.29: Отправление данных на github

```
MIGolovina_os-intro [Работает] - Oracle VM VirtualBox
Файл Машина Вид Ввод Устройства Справка
C6, 1 марта 19:28
migolovina@migolovina:~/git-extended
- Release branch 'release/1.0.0' has been locally deleted
- You are now on branch 'develop'

migolovina@migolovina:~/git-extended$ git push --all
Перечисление объектов: 6, готово.
Подсчет объектов: 100% (6/6), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (4/4), готово.
Запись объектов: 100% (5/5), 533 байта | 533.00 КиБ/с, готово.
Total 5 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 1 local object.
To github.com:migolovina/git-extended.git
   ab409db..e5b78f0  develop -> develop
   ab409db..c5c730d  main -> main
migolovina@migolovina:~/git-extended$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 168 байтов | 168.00 КиБ/с, готово.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:migolovina/git-extended.git
 * [new tag]         1.0.0 -> 1.0.0
migolovina@migolovina:~/git-extended$ gh release create v1.0.0 -F CHANGELOG.md
https://github.com/migolovina/git-extended/releases/tag/v1.0.0
migolovina@migolovina:~/git-extended$
```

Рис. 4.30: Создание релиза на github

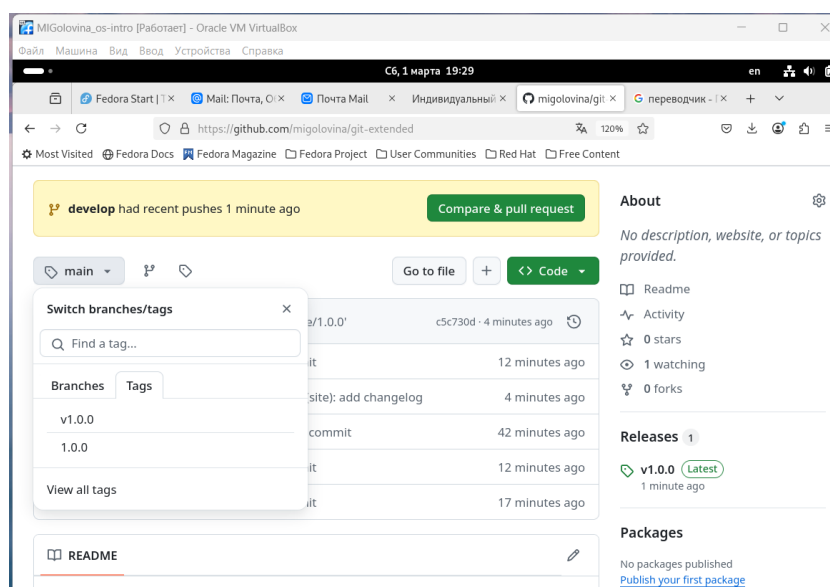


Рис. 4.31: Проверка

12. Работа с репозиторием git. (рис. 4.32-4.40).

The screenshot shows a terminal window titled "migolovina@migolovina:~/git-extended". The terminal output includes the following commands and their results:

```
To github.com:migolovina/git-extended.git
ab409db..e5b78f0 develop -> develop
ab409db..c5c730d main -> main
migolovina@migolovina:~/git-extended$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 168 байтов | 168.00 КиБ/с, готово.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:migolovina/git-extended.git
 * [new tag]      1.0.0 -> 1.0.0
migolovina@migolovina:~/git-extended$ gh release create v1.0.0 -F CHANGELOG.md
https://github.com/migolovina/git-extended/releases/tag/v1.0.0
migolovina@migolovina:~/git-extended$ git flow feature start feature_branch
Переключились на новую ветку «feature/feature_branch»

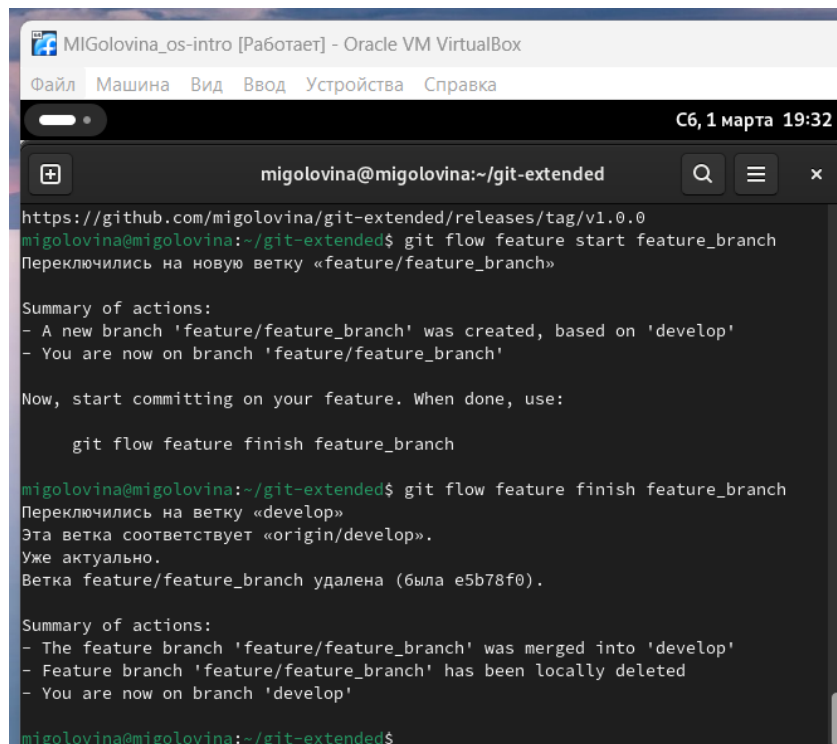
Summary of actions:
- A new branch 'feature/feature_branch' was created, based on 'develop'
- You are now on branch 'feature/feature_branch'

Now, start committing on your feature. When done, use:

    git flow feature finish feature_branch

migolovina@migolovina:~/git-extended$
```

Рис. 4.32: Создание ветки для новой функциональности



The screenshot shows a terminal window titled "migolovina@migolovina:~/git-extended". The terminal output is as follows:

```
https://github.com/migolovina/git-extended/releases/tag/v1.0.0
migolovina@migolovina:~/git-extended$ git flow feature start feature_branch
Переключились на новую ветку «feature/feature_branch»

Summary of actions:
- A new branch 'feature/feature_branch' was created, based on 'develop'
- You are now on branch 'feature/feature_branch'

Now, start committing on your feature. When done, use:

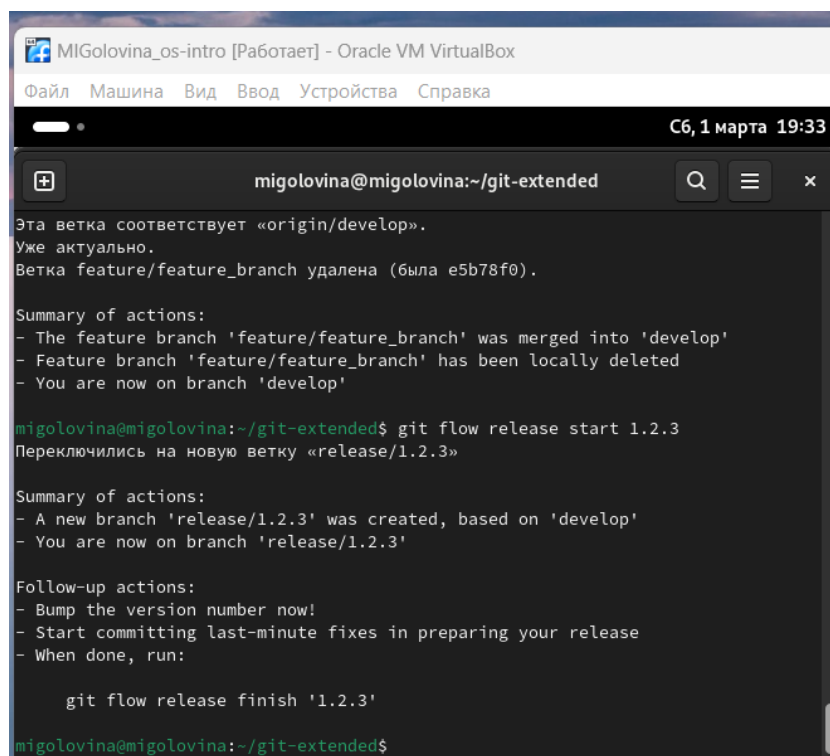
    git flow feature finish feature_branch

migolovina@migolovina:~/git-extended$ git flow feature finish feature_branch
Переключились на ветку «develop»
Эта ветка соответствует «origin/develop».
Уже актуально.
Ветка feature/feature_branch удалена (была e5b78f0).

Summary of actions:
- The feature branch 'feature/feature_branch' was merged into 'develop'
- Feature branch 'feature/feature_branch' has been locally deleted
- You are now on branch 'develop'

migolovina@migolovina:~/git-extended$
```

Рис. 4.33: Объединение ветки feature_branch с develop



МIGolovina_os-intro [Работает] - Oracle VM VirtualBox

Файл Машина Вид Ввод Устройства Справка

С6, 1 марта 19:33

migolovina@migolovina:~/git-extended

```
Эта ветка соответствует «origin/develop».
Уже актуально.
Ветка feature/feature_branch удалена (была e5b78f0).

Summary of actions:
- The feature branch 'feature/feature_branch' was merged into 'develop'
- Feature branch 'feature/feature_branch' has been locally deleted
- You are now on branch 'develop'

migolovina@migolovina:~/git-extended$ git flow release start 1.2.3
Переключились на новую ветку «release/1.2.3»


Summary of actions:
- A new branch 'release/1.2.3' was created, based on 'develop'
- You are now on branch 'release/1.2.3'

Follow-up actions:
- Bump the version number now!
- Start committing last-minute fixes in preparing your release
- When done, run:

    git flow release finish '1.2.3'

migolovina@migolovina:~/git-extended$
```


Рис. 4.34: Создание релиза с версией 1.2.3:



МIGolovina_os-intro [Работает] - Oracle VM VirtualBox

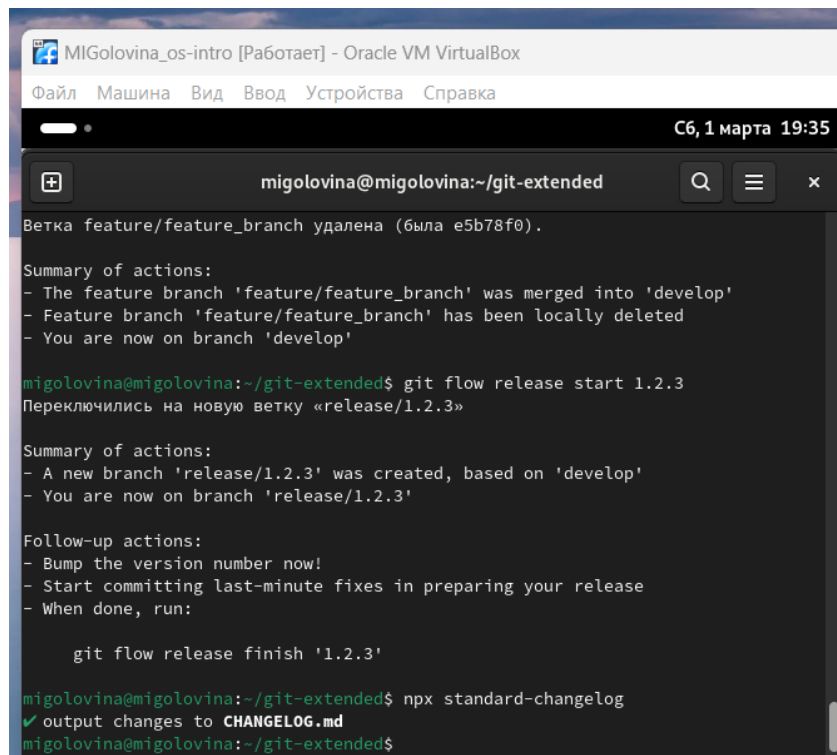
Файл Машина Вид Ввод Устройства Справка

С6, 1 марта 19:34

Открыть  package.json
~/git-extended

```
{
  "name": "git-extended",
  "version": "1.2.3",
  "description": "Git repo for educational purposes",
  "main": "index.js",
  "repository": "git@github.com:migolovina/git-extended.git",
  "author": "migolovina <sasha.alexanderqwe.ivanov@gmail.com>",
  "license": "CC-BY-4.0",
  "config": {
    "commitizen": {
      "path": "cz-conventional-changelog"
    }
  }
}
```

Рис. 4.35: Обновление номера версии в файле package.json.



The screenshot shows a terminal window titled "migolovina@migolovina:~/git-extended". The window displays the output of several Git commands. It starts with a message indicating that the 'feature/feature_branch' has been merged into 'develop' and then deleted. The user then runs 'git flow release start 1.2.3', which creates a new 'release/1.2.3' branch. The terminal shows the 'Summary of actions' and 'Follow-up actions' for this process. Finally, the user runs 'npx standard-changelog', which generates a 'CHANGELOG.md' file.

```
migolovina@migolovina:~/git-extended$ git flow release start 1.2.3
Переключились на новую ветку «release/1.2.3»

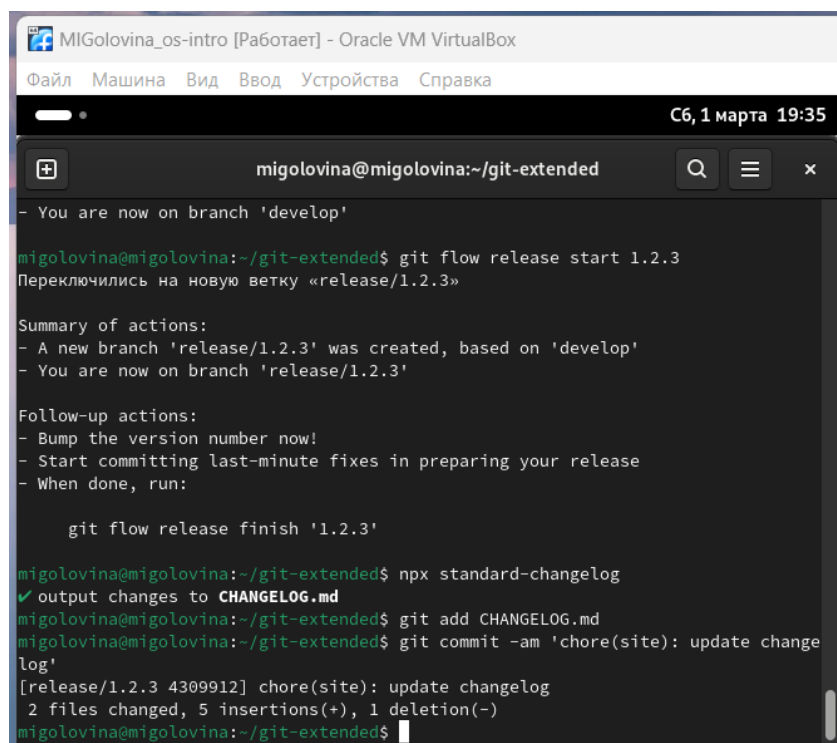
Summary of actions:
- A new branch 'release/1.2.3' was created, based on 'develop'
- You are now on branch 'release/1.2.3'

Follow-up actions:
- Bump the version number now!
- Start committing last-minute fixes in preparing your release
- When done, run:

    git flow release finish '1.2.3'

migolovina@migolovina:~/git-extended$ npx standard-changelog
✓ output changes to CHANGELOG.md
migolovina@migolovina:~/git-extended$
```

Рис. 4.36: Создание журнала изменений



```
MIGolovina_os-intro [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
C6, 1 марта 19:35
migolovina@migolovina:~/git-extended
- You are now on branch 'develop'
migolovina@migolovina:~/git-extended$ git flow release start 1.2.3
Переключились на новую ветку «release/1.2.3»

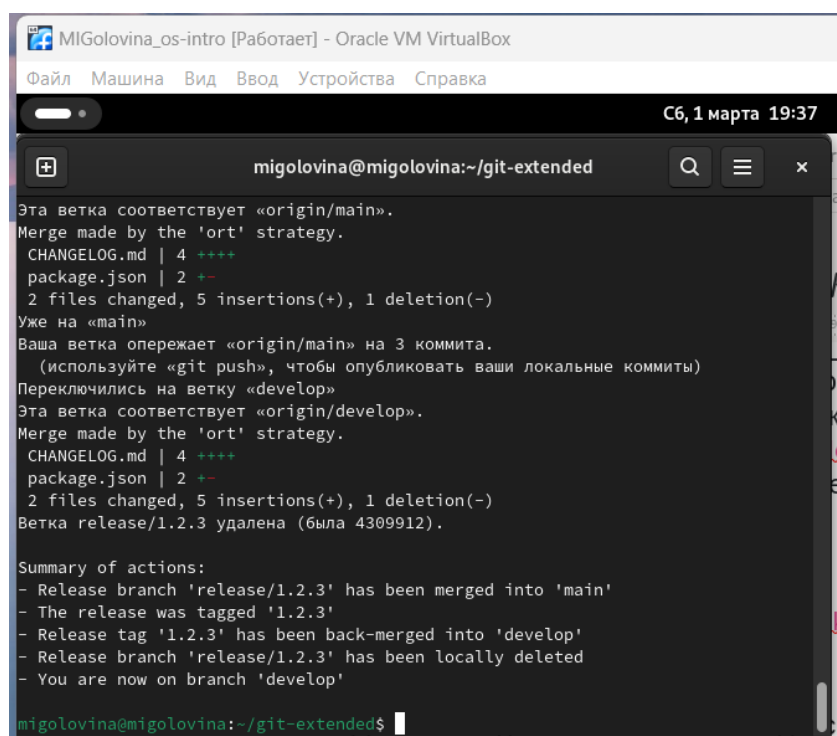
Summary of actions:
- A new branch 'release/1.2.3' was created, based on 'develop'
- You are now on branch 'release/1.2.3'

Follow-up actions:
- Bump the version number now!
- Start committing last-minute fixes in preparing your release
- When done, run:

    git flow release finish '1.2.3'

migolovina@migolovina:~/git-extended$ npx standard-changelog
✓ output changes to CHANGELOG.md
migolovina@migolovina:~/git-extended$ git add CHANGELOG.md
migolovina@migolovina:~/git-extended$ git commit -am 'chore(site): update changelog'
[release/1.2.3 4309912] chore(site): update changelog
2 files changed, 5 insertions(+), 1 deletion(-)
migolovina@migolovina:~/git-extended$
```

Рис. 4.37: Добавление журнала изменений в индекс



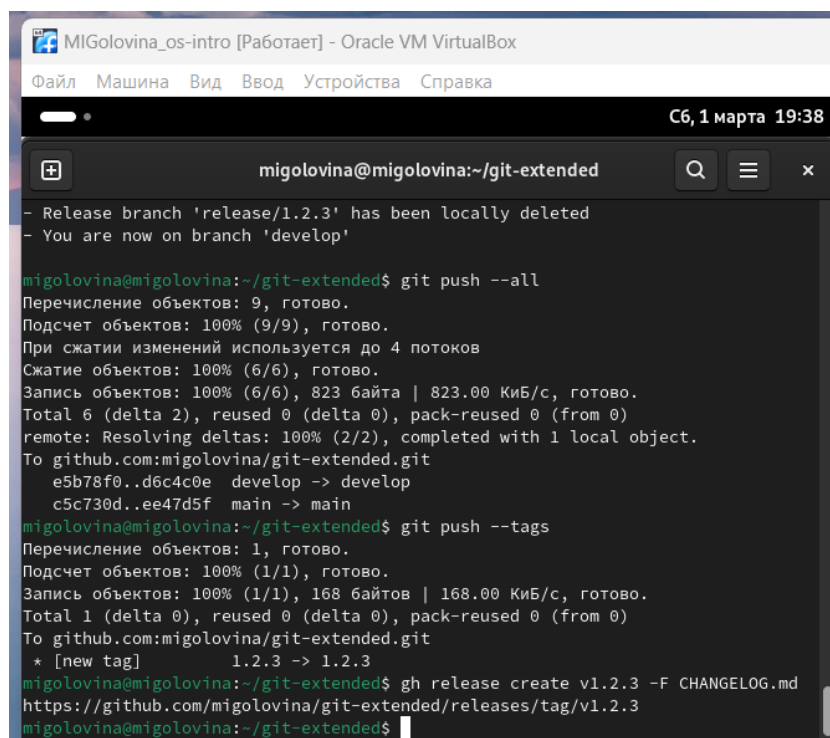
The screenshot shows a terminal window titled "migolovina@migolovina:~/git-extended". The terminal output displays the result of a Git merge operation. It starts with a confirmation that the branch corresponds to "origin/main", followed by a merge made by the 'ort' strategy. A diff shows changes to CHANGELOG.md and package.json. The terminal then indicates the current branch is 'main' and that the local branch is ahead of 'origin/main' by 3 commits. It prompts the user to use 'git push' to publish local commits. Next, it switches to the 'develop' branch and shows another merge by the 'ort' strategy with the same file changes. Finally, it reports that the 'release/1.2.3' branch has been deleted. A summary of actions follows, detailing the merge into 'main', tagging '1.2.3', back-merging into 'develop', and deleting the release branch. The terminal ends with the prompt "migolovina@migolovina:~/git-extended\$".

```
Эта ветка соответствует «origin/main».
Merge made by the 'ort' strategy.
  CHANGELOG.md | 4 ++++
  package.json | 2 +-
  2 files changed, 5 insertions(+), 1 deletion(-)
Уже на «main»
Ваша ветка опережает «origin/main» на 3 коммита.
  (используйте «git push», чтобы опубликовать ваши локальные коммиты)
Переключились на ветку «develop»
Эта ветка соответствует «origin/develop».
Merge made by the 'ort' strategy.
  CHANGELOG.md | 4 ++++
  package.json | 2 +-
  2 files changed, 5 insertions(+), 1 deletion(-)
Ветка release/1.2.3 удалена (была 4309912).

Summary of actions:
- Release branch 'release/1.2.3' has been merged into 'main'
- The release was tagged '1.2.3'
- Release tag '1.2.3' has been back-merged into 'develop'
- Release branch 'release/1.2.3' has been locally deleted
- You are now on branch 'develop'

migolovina@migolovina:~/git-extended$
```

Рис. 4.38: Залитие релизной ветки в основную ветку



```
migolovina@migolovina:~/git-extended$ git push --all
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 823 байта | 823.00 КиБ/с, готово.
Total 6 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To github.com:migolovina/git-extended.git
   e5b78f0..d6c4c0e  develop -> develop
   c5c730d..ee47d5f  main -> main
migolovina@migolovina:~/git-extended$ git push --tags
Перечисление объектов: 1, готово.
Подсчет объектов: 100% (1/1), готово.
Запись объектов: 100% (1/1), 168 байтов | 168.00 КиБ/с, готово.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To github.com:migolovina/git-extended.git
 * [new tag]           1.2.3 -> 1.2.3
migolovina@migolovina:~/git-extended$ gh release create v1.2.3 -F CHANGELOG.md
https://github.com/migolovina/git-extended/releases/tag/v1.2.3
migolovina@migolovina:~/git-extended$
```

Рис. 4.39: Отправление данных на github и создание релиза на github с комментарием из журнала изменений

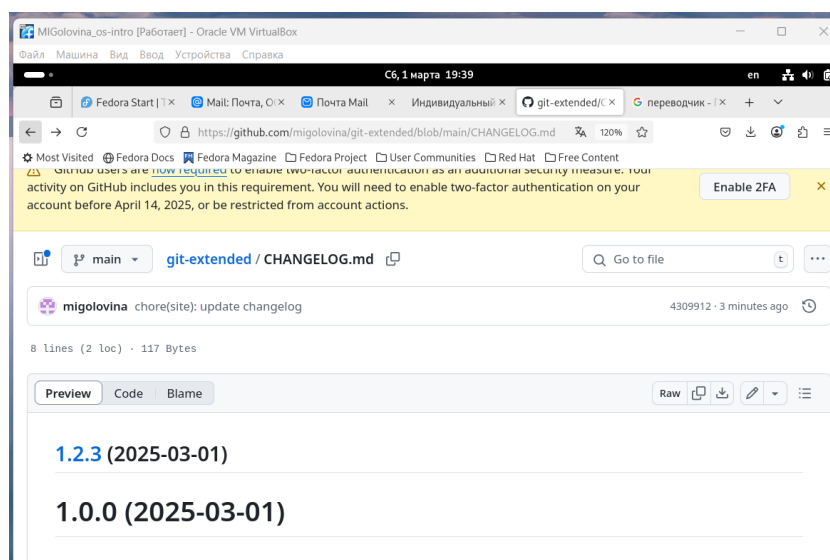


Рис. 4.40: Проверка релизов

5 Выводы

Я получила навыки правильной работы с репозиториями git.

Список литературы

1. Dash, P. Getting Started with Oracle VM VirtualBox / P. Dash. – Packt Publishing Ltd, 2013. – 86 сс.
2. Colvin, H. VirtualBox: An Ultimate Guide Book on Virtualization with VirtualBox. VirtualBox / H. Colvin. – CreateSpace Independent Publishing Platform, 2015. – 70 сс.
3. Vugt, S. van. Red Hat RHCSA/RHCE 7 cert guide : Red Hat Enterprise Linux 7 (EX200 and EX300) : Certification Guide. Red Hat RHCSA/RHCE 7 cert guide / S. van Vugt. – Pearson IT Certification, 2016. – 1008 сс.
4. Робачевский, А. Операционная система UNIX / А. Робачевский, С. Немнюгин, О. Стесик. – 2-е изд. – Санкт-Петербург : БХВ-Петербург, 2010. – 656 сс.
5. Немет, Э. Unix и Linux: руководство системного администратора. Unix и Linux / Э. Немет, Г. Снайдер, Т.Р. Хейн, Б. Уэйли. – 4-е изд. – Вильямс, 2014. – 1312 сс.
6. Колисниченко, Д.Н. Самоучитель системного администратора Linux : Системный администратор / Д.Н. Колисниченко. – Санкт-Петербург : БХВ-Петербург, 2011. – 544 сс.
7. Robbins, A. Bash Pocket Reference / A. Robbins. – O'Reilly Media, 2016. – 156 сс.