

Git в DevOps

4 марта 2026 г.



Name

- v1.0
- v2.0
- v2.1
- v2.2

- project
- project-revised
- project-final
- project-final-for-real

Name

- project
- projectt
- projecttttttt
- aaaaaaaaaaaaaaaaaaaaaaaa...

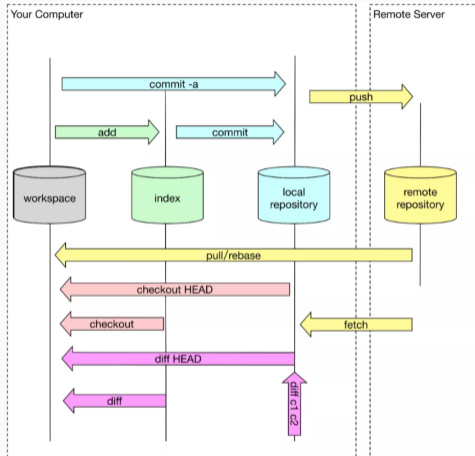


Git — формат хранения истории проекта и правила работы с этой историей.

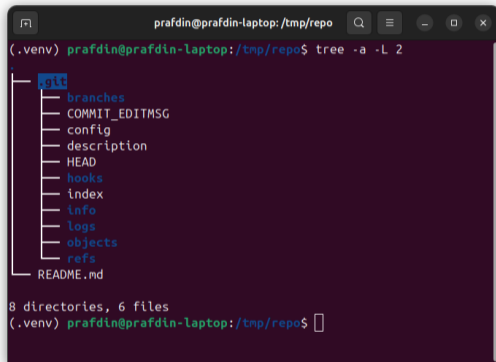
Git репозиторий — проект, управляемый Git, с сохранённой историей изменений.

Git CLI — утилита командной строки для работы с Git репозиторием.

Git как система



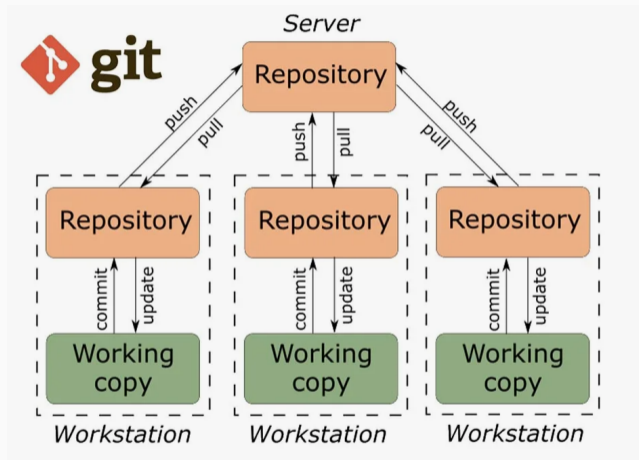
<https://www.alibabacloud.com/blog/a-detailed-exp...>



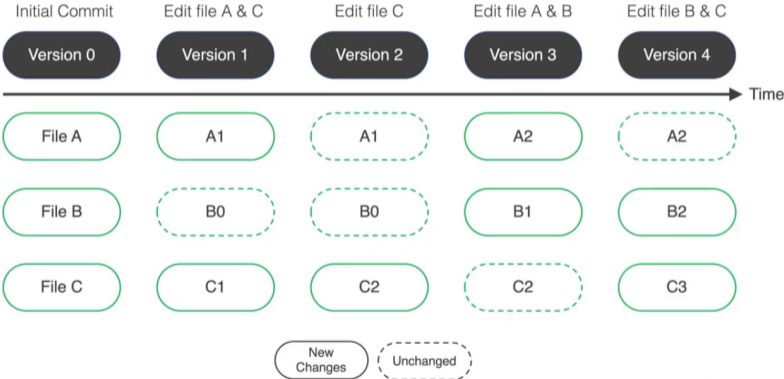
```
prafdin@prafdin-laptop: /tmp/repo
(.venv) prafdin@prafdin-laptop: /tmp/repo$ tree -a -L 2
.
├── .git
│   ├── branches
│   ├── COMMIT_EDITMSG
│   ├── config
│   ├── description
│   ├── HEAD
│   ├── hooks
│   ├── index
│   ├── info
│   ├── logs
│   ├── objects
│   └── refs
└── README.md

8 directories, 6 files
(.venv) prafdin@prafdin-laptop: /tmp/repo$
```

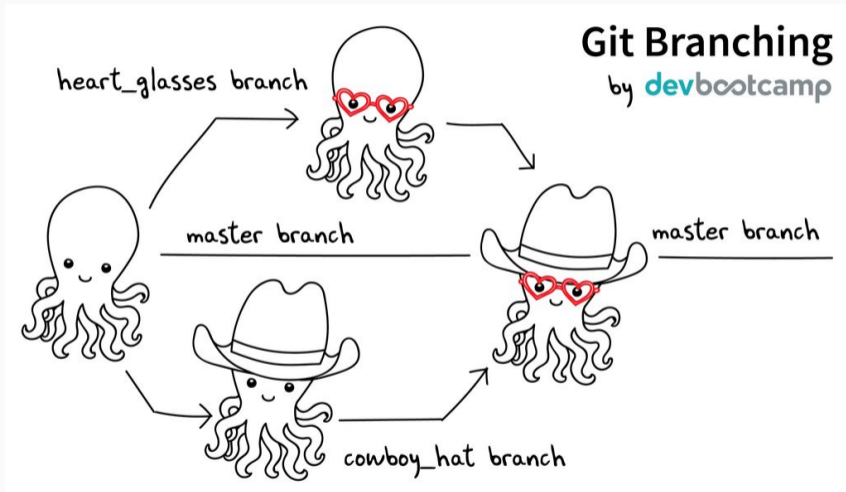
Демонстрация



Version control with Git



Adapted from <https://git-scm.com/book/en/v2>



GitHub — сервис для хранения и управления Git-репозиториями с дополнительными возможностями для совместной работы над проектом.

Возможности:

1. Общее пространство для совместной работы над слиянием кода (pull requests).
2. Управление списком задач и проблем проекта (issues).
3. Обсуждение и коммуникация (discussions).
4. Платформа автоматизации (GitHub Actions).
5. Аналитика проекта.

Git ранее

- Хранилище исходного кода
- Версионирование файлов
- Работа с ветками
- Совместная разработка

Git в DevOps

- Центр управления системой
- Source of Truth для системы
- Триггер автоматизации
- Аудит изменений

Git репозиторий становится местом хранения состояния **всей системы**

Что хранится в DevOps репозитории?

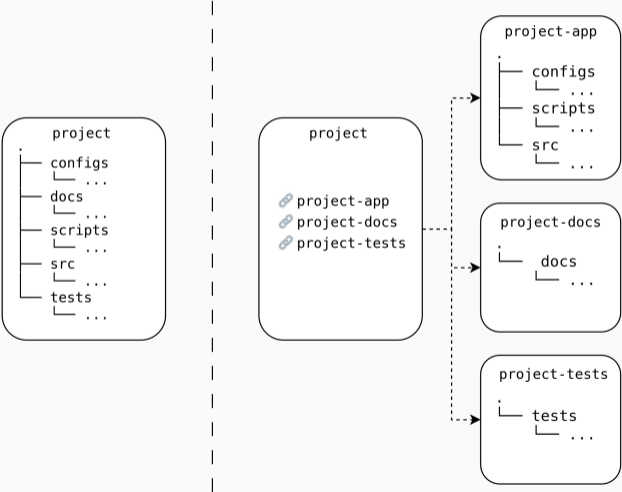
my-project/

- src/ — исходный код приложения
- config/ — файлы конфигурации
- scripts/ — скрипты развертывания
- docs/ — документация и инструкции
- tests/ — тесты и проверки

Принцип: если это влияет на систему — это должно быть в Git

Monorepo vs Polyrepo

Monorepo vs Polyrepo



События в Git запускают процессы

Традиционный подход:

- Ручное развертывание
- Разделенные процессы
- Человеческий фактор
- Нет связи код <-> система

DevOps подход:

- git push → автодеплой
- git tag → релиз
- Pull Request → тестирование
- Коммит = событие системы

Принцип: каждое изменение в Git влияет на всю систему

Что покажем:

- Настройка GitHub webhook
- Настройка доступа к VM через реверс прокси
- Автоматический запуск деплоя по событию в репозитории

Демо-репозиторий:

<https://github.com/prafdin/devops-demo-website/tree/webhook-demo>

- <https://git-scm.com/book/en/v2>
- https://learning.nceas.ucsb.edu/2024-10-coreR/session_07.html