

Git Strategy

Стратегия Gitflow Workflow использует центральный репозиторий в качестве обеспечения связи для всех разработчиков. И, как и в других рабочих процессах, разработчики работают на локальном уровне и производят слияние ветви с центральным репозиторием(рис. 1).

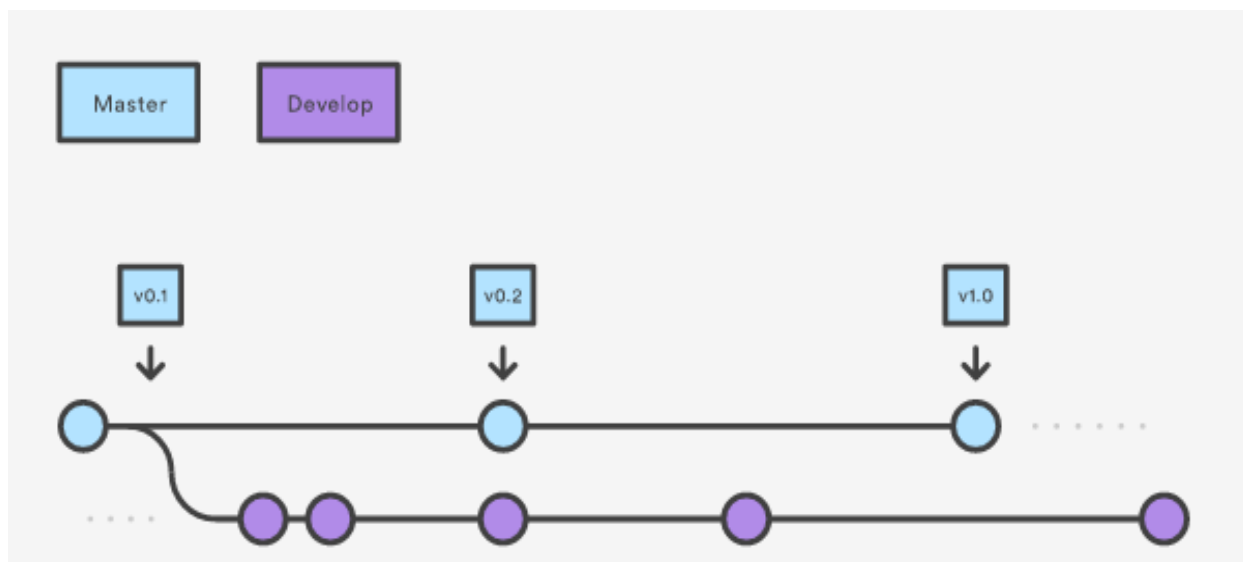
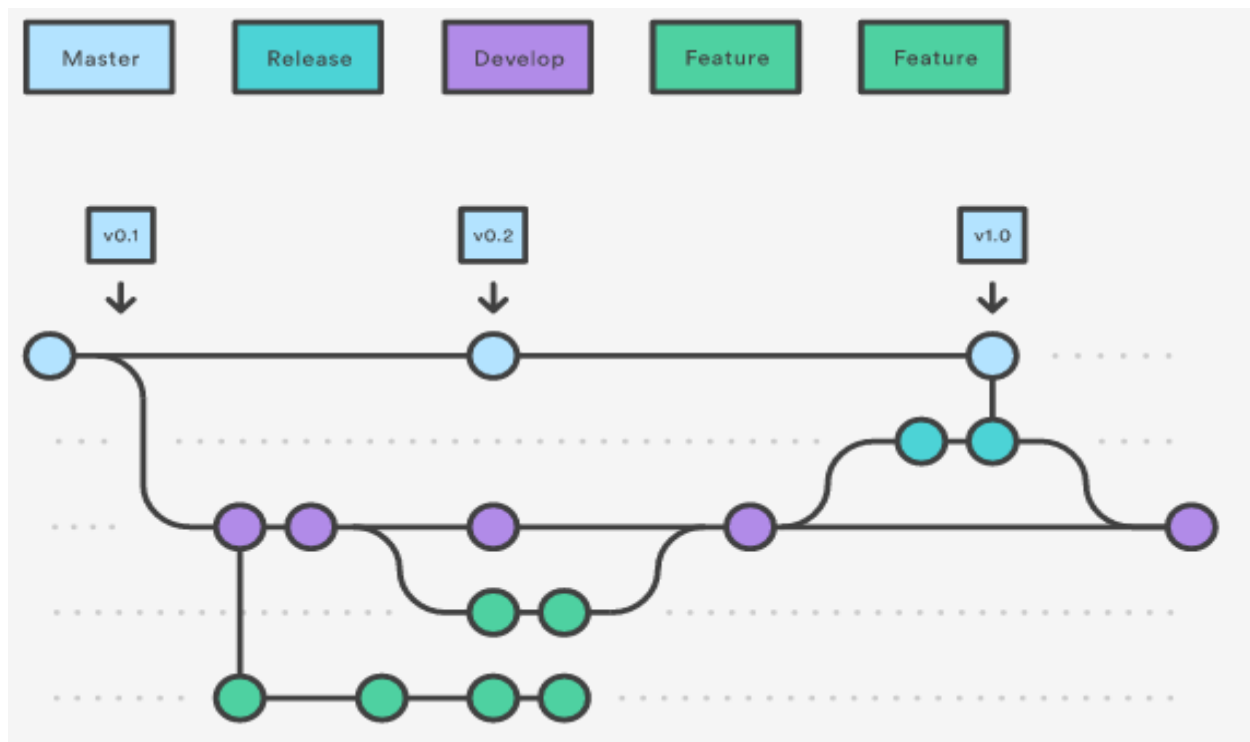


Рис.1

Вместо одной master-branch, этот рабочий процесс использует две ветви, чтобы вести историю проекта. Главная ветвь (master) содержит историю комитов разных релизов продукта, в то время как develop branch служит хранилищем релизов всех дополнительных функций.

Feature Branches

Каждая новая функция должна находиться в своей ветке. Но, вместо того, чтобы разветвлять ветку мастера, feature branch используют develop branch в качестве ветки для слияния. Когда функционал доработан, он будет merge обратно в ветку develop. Feature branch не должен взаимодействовать непосредственно с master branch.



После того как develop branch обрела достаточно функционала для релиза (или заранее определенная дата релиза приближается), создаётся новая ветвь с готовым функционалом для релиза и соединяется с master branch. Создание этой ветки начинает следующий цикл разработки, в это время никакой новый функционал не может быть добавлен. С этого момента в новой ветке выполняется исправления ошибок, создание документации, и выполнение других релиз-ориентированных задач. После того, как все будет готово, релиз сливается с master branch и помечается версия. Кроме того, данная ветка должна быть обратно merge с веткой develop, которая могла меняться за время внедрения нового функционала.

Использование дополнительной ветки для подготовки релизов позволяет одной команде максимально доработать текущий релиз, в то время как другая команда продолжает работать над функциями для следующего релиза. Данная ветка также создает четко определенные фазы развития .

Общие соглашения:

- branch off: develop
- merge into: master
- naming convention: release-* or release/*