



# git



Semifir



# Workflow



# Workflow

Bonnes pratiques <http://nvie.com/posts/a-successful-git-branching-model>

Le but de ce système est de séparer les branches et les différentes versions du projet.

## Les branches éternelles

Les branches **master** et **develop** sont les seules branches qui ne seront jamais supprimées pendant toute la durée du projet.

### La branche de production (master)

1. les versions publiées pour les utilisateurs (nouvelles fonctionnalités + correctif).
2. Chaque commit de cette branche reçoit un Tag de version

branches qui peuvent être créées à partir de master :

1. la branche develop (une seule fois au début du projet)
2. les branches de correctifs (hotfix).

branches dont master pourra recevoir les modifications :

1. les branches de nouvelles versions (release),
2. les branches de correctifs (hotfix).



## Les branches éternelles

### La branche de développement (develop)

1. recevra toutes les nouvelles fonctionnalités qui ne sont pas encore intégrées à la version principale. futures versions qui seront publiées et considérée comme instable.
2. Cette branche est créée dès le début du projet à partir de la branche master.

branches pouvant être créées à partir de develop :

1. les branches de fonctionnalités (feature),
2. les branches de versions (release).

develop pourra recevoir les modifications depuis :

1. les branches de correctifs (hotfix),
2. les branches de versions (release),
3. les branches de fonctionnalités (feature).

## Les branches éphémères

créées dans un but très précis, une fois accompli, ces branches sont supprimées.

## Les branches de versions (release)

versions Beta du projet

Exemple : La branche release-2.0 recevra les correctifs spécifiques aux nouvelles fonctionnalités

Aucune fonctionnalité de la version 3.0. Elle est créée à partir de develop

Cette branche sera mergée dans les branches develop et master avec le tag v2.0 et sera supprimée

## Les branches de correctifs (hotfix)

correctifs destinés à la production lorsqu'un bug est découvert et doit être corrigé.

La branche est créée à partir de la branche master puis sera mergée/supprimé dans master et develop

## Les branches éphémères

créées dans un but très précis, une fois accompli, ces branches sont supprimées.

## Les branches de fonctionnalités (feature)

- intégrer une nouvelle fonctionnalité à la branche develop.
- branches sont locales (non partagées)
- Une intégrée dans develop, elle est supprimée

# Git flow





