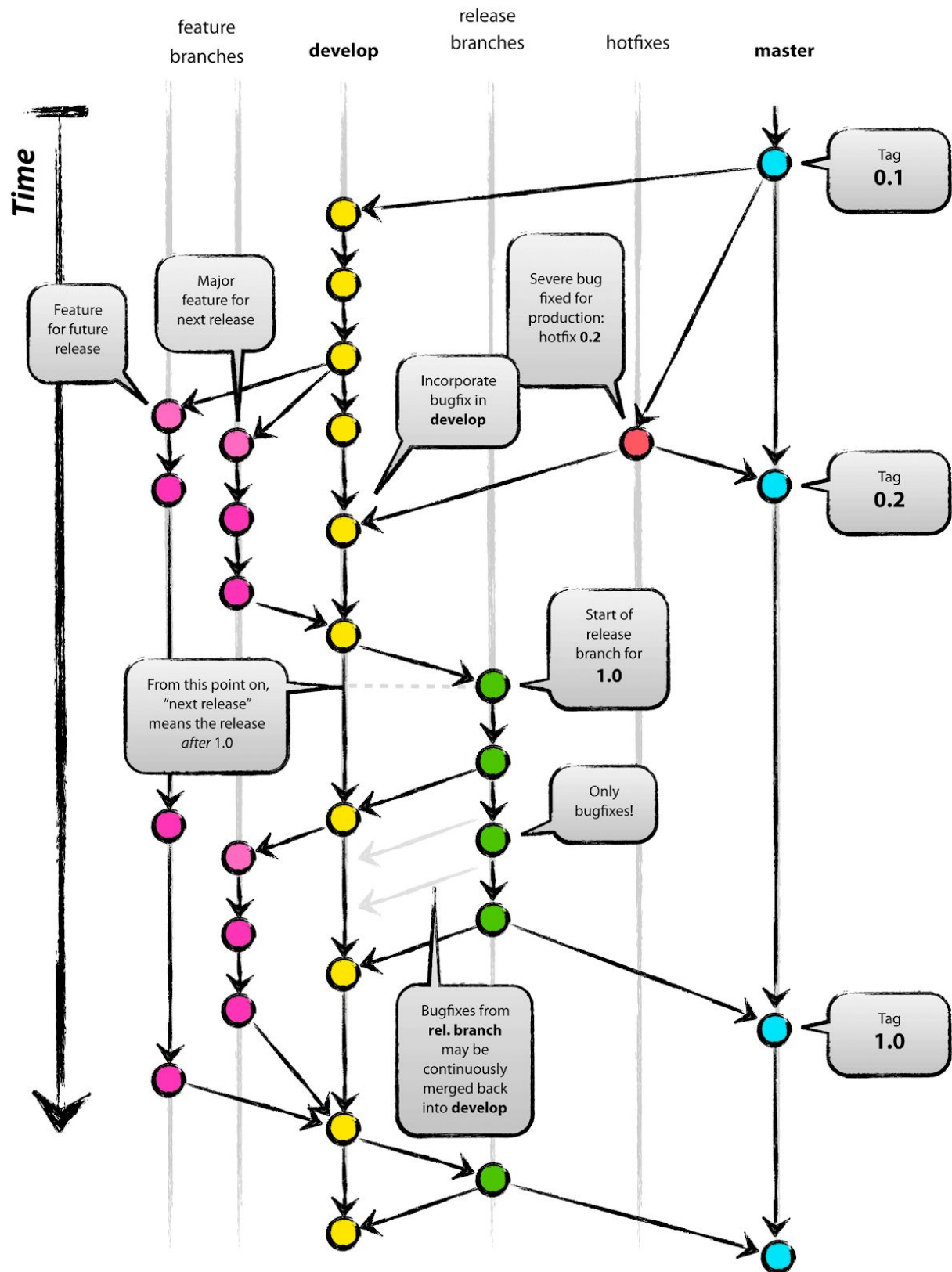


GIT FLOW PROCESS



Các thành phần chính trong gitflow

Gitflow sẽ chia toàn bộ quá trình làm việc ra làm 2 phần chính:

- Main Branches
 - Master Branch: Nơi lưu trữ đại diện cho những phiên bản (public - production) của dự án
 - Develop Branch: Là nơi lưu trữ(tích hợp) các chức năng(feats) hay sửa lỗi(fix)... của phiên bản tiếp theo trước khi được đưa đi lưu trữ tại Master Branch
- Supporting Branches
 - Feature branches : khi khởi tạo(init) 1 gitflow thì tất cả các feature được tạo sẽ được đưa vào trong thư mục feature(cá nhân của từng người).
 - Release branches : nơi lưu trữ các bản “demo” của bản production trước khi chính thức được merge vào master
 - Hotfix branches : khi cần xử lý nhanh những vấn đề được phát trên branch Master thì cần tạo 1 hotfix để xử lý vấn đề. Lưu ý trong trường hợp này không tạo feature/..

Chi tiết các thành phần và cách hoạt động

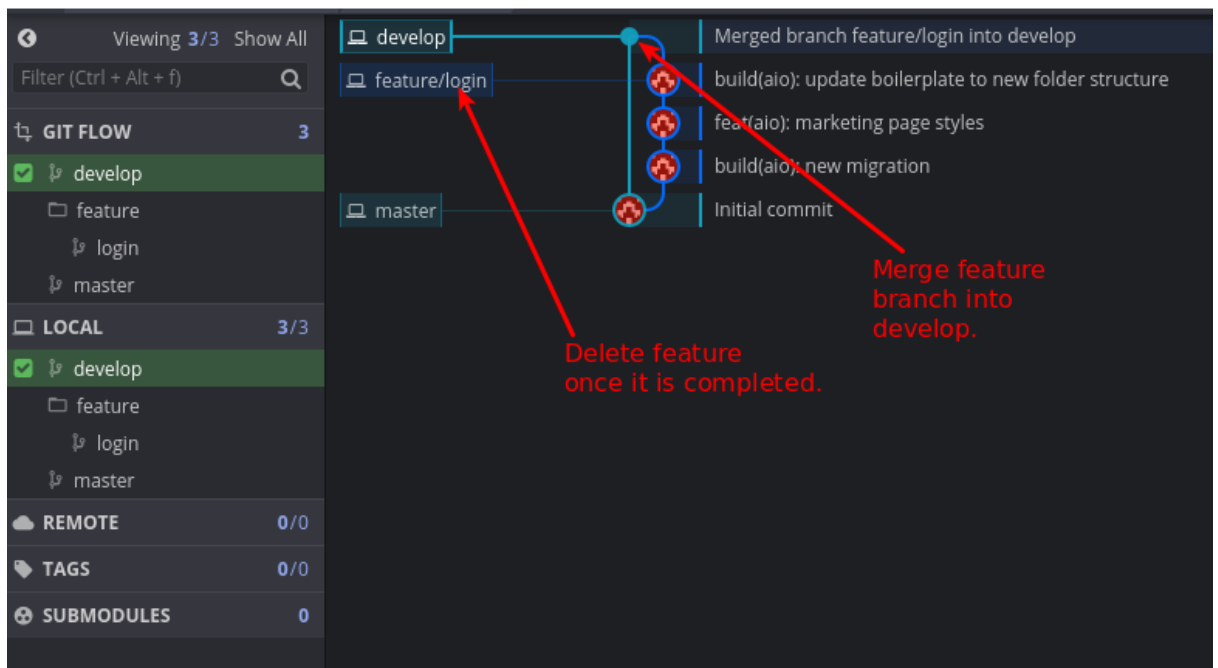
- Master Branch và Develop Branch chỉ là nơi lưu kết quả nên gần như không có thao tác commit trực tiếp vào 2 branch này.
- Chỉ có duy nhất 2 branch : Develop và Master, không tạo Develop 1,....



- Feature Branches
 - “Có thể” được tạo ra từ branch Develop.
 - “Bắt buộc” phải merge lại branch Develop khi kết thúc.
 - Convention Name: feature/*
 - No tag.
 - Chỉ được merge vào branch develop khi feature hoàn thành và ổn định.
 - Thông thường, 1 feature sẽ chỉ tồn tại trong repo riêng của từng develop cho đến khi nó được hoàn thành. Để tránh trường hợp git

chứa quá nhiều commit thì developer có thể squash lại trước khi commit hoặc trước khi merge(merge request).

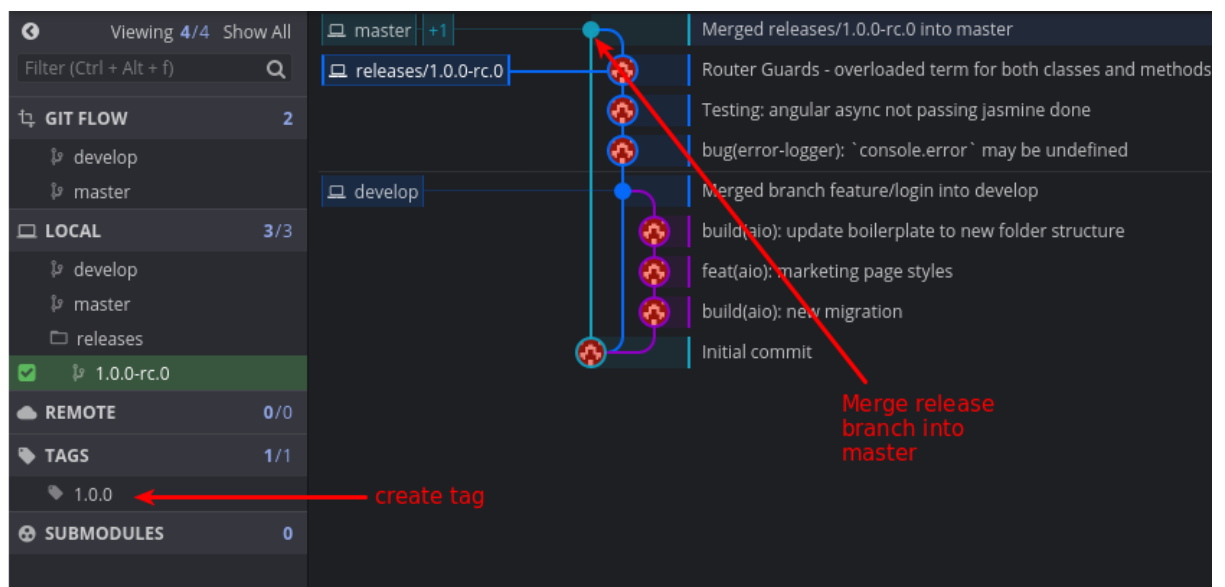
- Xóa các feature khi nó kết thúc.



● Release Branch

- “Có thể” được tạo ra từ Develop
- “Bắt buộc” phải được merge vào Develop và Master
- **Convention Name: release/<future-master version> - [optional]<release name>**
- **Tag : version name = tag future Master version <optional>**
- Khi số tính năng đã “tương đối đủ” cho giai đoạn phát hành(master) hoặc đã gần ngày release phiên bản, cần tách việc phát triển khỏi nhánh develop và tạo 1 nhánh release.

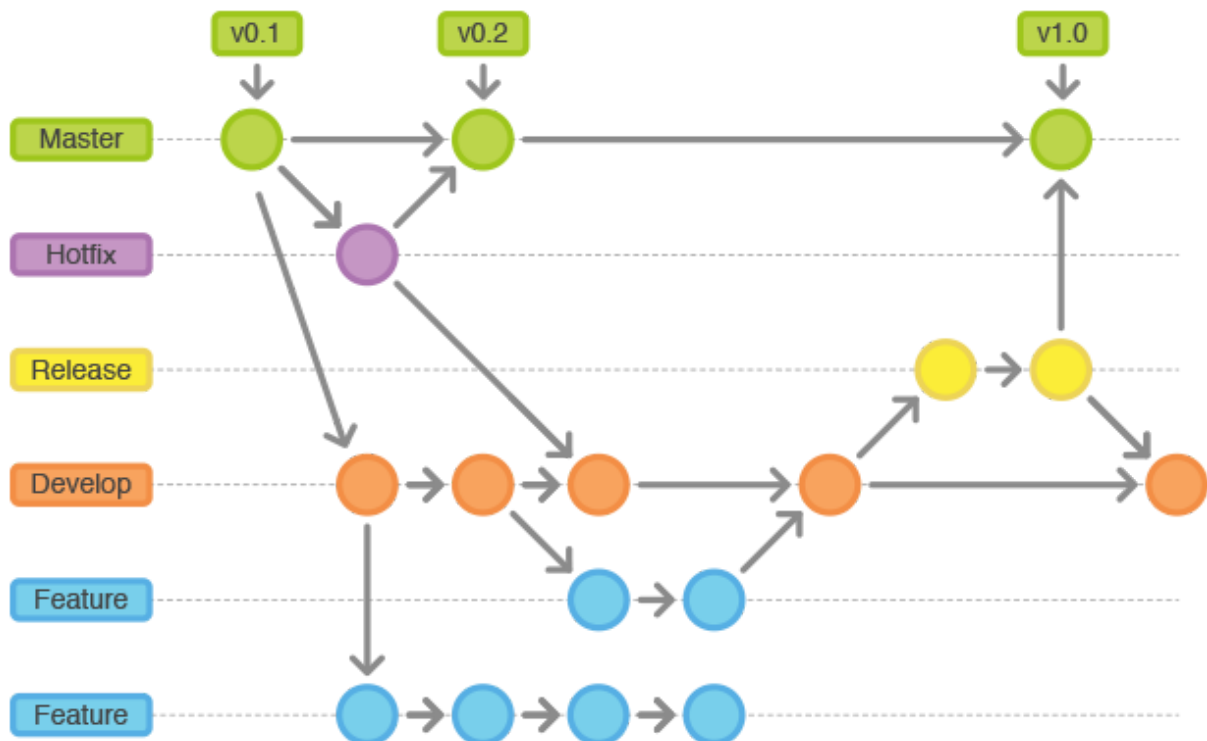
- Tại thời điểm tạo branch release, sẽ không thêm tính năng mới mà chỉ fix bug và bổ sung tài liệu cho đến khi branch release hoàn thiện.
- Giai đoạn kết thúc 1 release, có thể thực hiện theo các bước:
 - Merge release vào master và gắn tag cho master
 - Merge release vào develop
 - Xoá release version vừa rồi(optional)

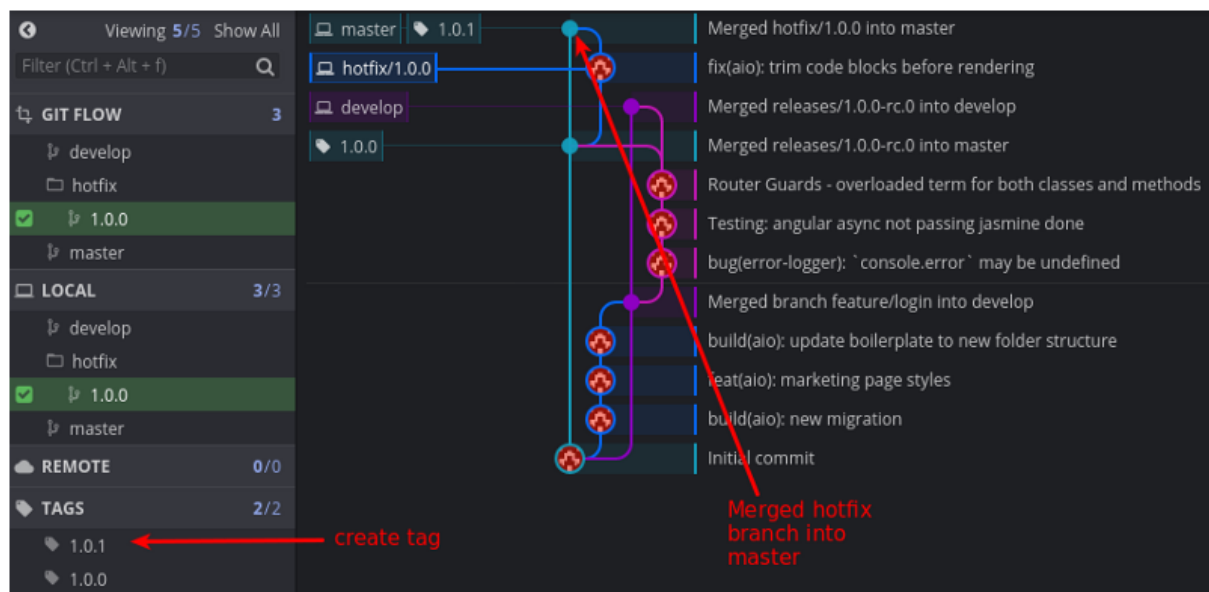
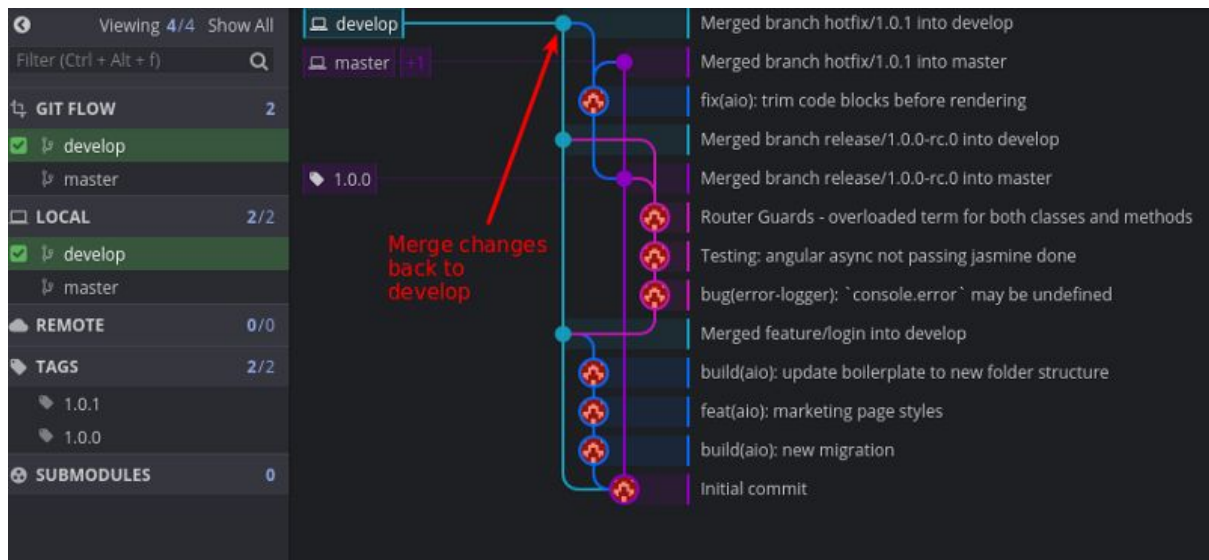




- **Hotfix Branch(Maintain Branch)**

- “Có thể” được tạo ra từ Master
- “Bắt buộc” phải được merge vào Develop và Master
- **Convention Name: hotfix/<version master> - <feature fix>**
- **Tag : bằng version của master version <optional>**
- Được tạo ra khi có bug phát sinh trên branch Master(thông thường do người dùng cuối phát hiện), version(tag) có thể tính theo version hiện tại của branch master. Sau khi fix xong bug, merge vào branch master và branch develop. Có thể xóa branch hotfix vừa tạo.





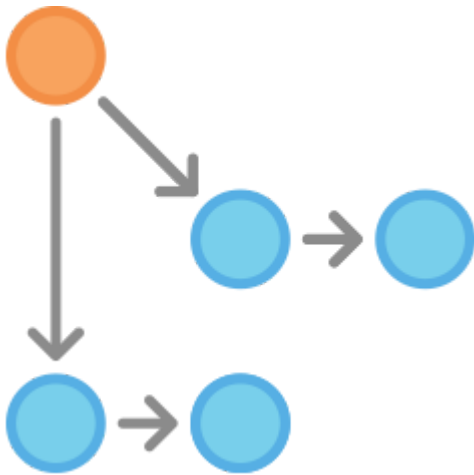
Chú ý. Để tránh việc version bị hỗn loạn, thì chỉ có version master cần được thống nhất quy tắc tăng, các tag(version) của các nhánh release, hotfix có thể lấy version master làm chuẩn và thêm các tiền tố tên để định danh.

Quy định về Role của các thành viên trong team

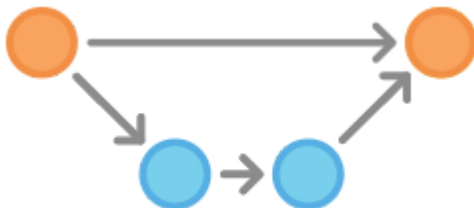
1. Tất cả các commit khi muốn merge vào 1 trong các nhánh: develop, release, hotfix, master đều cần tạo merge request.
2. Chỉ leader(maintainer) hoặc PM mới có quyền thực hiện thao tác merge trong các merge request.
3. Khi cần tạo branch mà ảnh hưởng trực tiếp tới develop và master cần thông qua leader team ví dụ như branch hotfix sẽ được merge vào master nên không được tạo 1 cách tùy tiện.

Một ví dụ về gitflow có thể mô tả như sau:

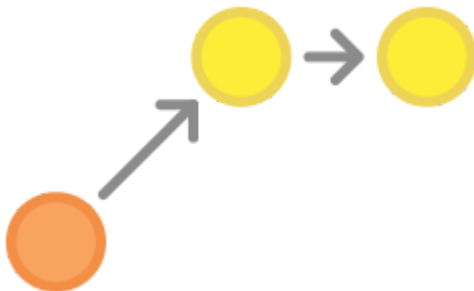
2 người A và B cùng bắt đầu các feature.



A hoàn thành feature và được merge vào develop.



A tạo 1 bản release từ phiên bản trên develop.



Kết thúc bản release tạo phiên bản trên master và merge code vào develop.

