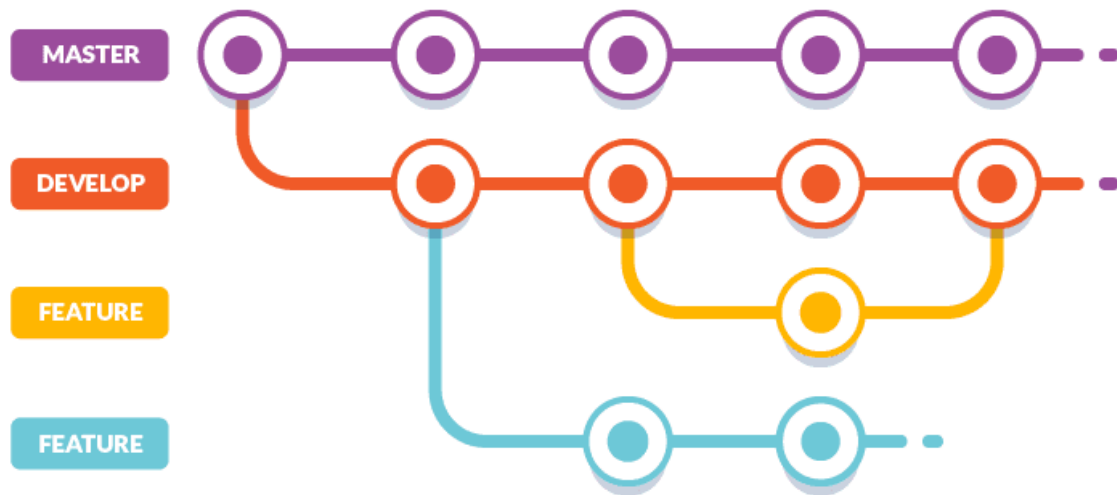


Nguyên tắc làm việc

1. Git workflow



2. Trello

Tạo task theo cú pháp: `[GS-<id>]: <Mô tả>`

ví dụ: `[GS-02]: Dockerize (backend)`

“

id tăng dần lên

Phân loại các tasks:

1. **Story**: Về non-technical (Phân tích, thiết kế, ...)
2. **Task**: Về technical (thêm tính năng, khởi tạo môi trường, ...)
3. **Bug**: Sửa lỗi hệ thống

Khi tạo task cần tối thiểu:

- Tên task
- Mô tả task
- Label gồm Priority và Type

5 Trạng thái:

- **Backlog**: Những tasks đang ở trạng thái chờ và sẽ được xét duyệt khi planning (sau khi duyệt thì task sẽ được assign cho người nào đó)
- **To-do**: Những task được assign và chưa bắt đầu làm

- **Doing** : Những task đang được làm
- **Testing** : Những task đang được kiểm thử
- **Done** : Những task hoàn tất

3. Nguyên tắc khi commit và pull requests

<https://datluu-1702.gitbook.io/todo-doc/dac-ta-co-so-du-lieu#1-quy-tac-comment>

a. Commit

[<Hành động>] : <thông tin>

- Hành động:

- **Add** = Thêm vào mã nguồn.
 - Ví dụ: chức năng, test, thư viện
- **Drop** = Xóa khỏi mã nguồn.
 - Ví dụ: chức năng, test, thư viện
- **Modified** = Sửa trong mã nguồn.
 - Ví dụ: chỉnh sửa, lỗi, typo
- **Bump** = Thay đổi version.
 - Ví dụ: nâng phiên bản một thư viện đang sử dụng
- **Make** = Thay đổi công cụ hoặc quy trình build liên quan hạ tầng
- **Refactor** = Sửa đổi nhằm mục đích tái cấu trúc mã nguồn cũ.
 - Ví dụ: Tách logic xử lý trong controller layer về business layer
- **Optimize** = Sửa đổi nhằm mục đích tối ưu hiệu năng cho mã nguồn cũ.
 - Ví dụ: Tối ưu hiệu năng chức năng tìm kiếm đơn hàng bằng cách sử dụng thêm caching layer để giảm thời gian truy vấn từ cơ sở dữ liệu.
- **Reformat** = Sửa đổi nhằm mục đích định dạng lại code cũ.
 - Ví dụ: xóa khoảng trắng, dòng trắng sai coding convention
- **Rephrase** = Sửa đổi liên quan tài liệu trong source code.
 - Ví dụ comment trong source code (TODO / FIXME / ...)
- **Document** = Sửa đổi liên quan đến tài liệu bên ngoài source code.
 - Ví dụ thêm mô tả vào file README.md

- Thông tin: đối tượng của hành động

Ví dụ:

[Add]: Packages & Initialization state

b. Pull requests

Title pull requests theo cú pháp: [<Type> | <GS-id>]: <Mô tả>

Ví dụ: [Feature | GS-02]: Dockerize backend

Type của pull requests bao gồm:

1. **Feature** : tương đương với **Task**
2. **Bugfix** : tương đương với **Bug**

3. **Document** : tương đương với **Story**

4. Nguyên tắc tạo tên branch

đứng ở branch **develop** và tạo branch theo cú pháp:

<Type of pull requests>/GS-id-mô-tả

Ví dụ:

Với task **[GS-02]: Dockerize (backend)**

thì tên branch là **Feature/GS-02-dockerize-backend**

“

Lưu ý: Mô tả theo tiếng anh

Ví dụ

Với task **[GS-03]: Viết tài liệu cơ bản về làm việc nhóm**

thì tên branch là **Document/GS-03-document-for-teamwork**