

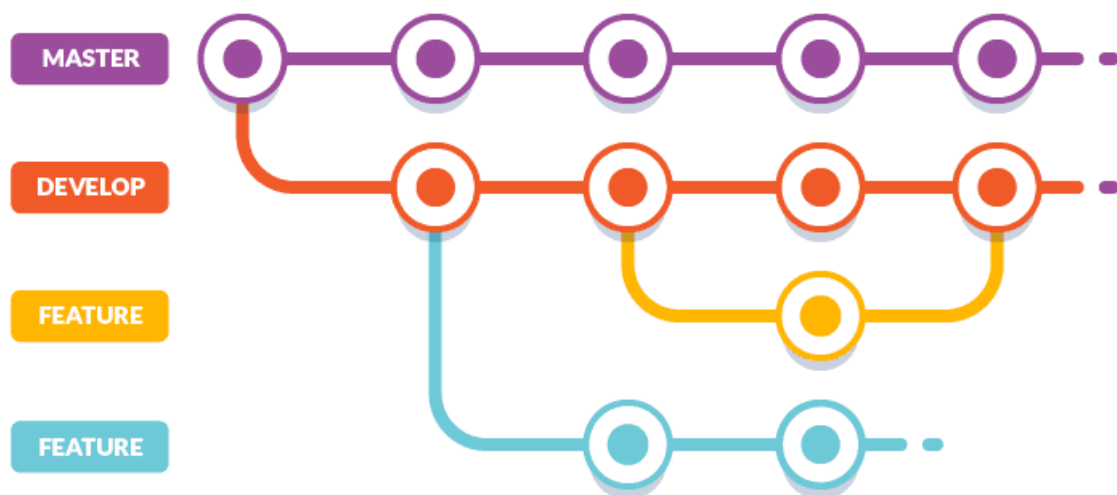
Nguyên tắc làm việc

Table of contents

1. Git workflow
 2. Trello workflow
-

Git workflow

1. Branch workflow



2. Nguyên tắc khi commit và pull requests

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>** | **<NP-id>**]: **<Mô tả>**

Ví dụ: [**Feature** | **NP-02**]: **Dockerize backend**

Type của pull requests bao gồm:

1. **Feature**: tương đương với **Task** trong Trello
2. **Bugfix**: tương đương với **Bug** trong Trello
3. **Document**: tương đương với **Story** trong Trello

Cú pháp pull request khi merge develop vào main/master branch

[**Release**] **Release version <version> to main branch**

3. Nguyên tắc tạo tên branch

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

<Type of pull requests>/NP-id-mô-tả

Ví dụ:

Với task `[NP-02]: Dockerize (backend)`

thì tên branch là `Feature/NP-02-dockerize-backend`

“

Lưu ý: Mô tả theo tiếng anh

Ví dụ

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

thì tên branch là `Document/NP-03-document-for-teamwork`

“

Khi pull request phải add thêm reviewer (leader và một người làm chung tech stack)

4. Lưu ý

1. Không bao giờ được đứng ở branch master và develop để code
2. Pull develop branch về trước khi push lên remote
3. Mọi conflict phải giải quyết ở local (trong trường hợp không biết chọn current change hay incoming change thì liên hệ với người code incoming change và leader)

Trello Workflow

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

ví dụ: `[NP-02]: Dockerize (backend)`

“

- id tăng dần lên
- Khi tạo chỉ cần nhập `mô tả`, còn `[NP-<id>]:` sẽ tự động tạo bằng `rule`
- `rule`: when a card is created in list "[31] Backlog" by anyone, rename the card to "[NP-{cardnumber}]: {cardname}"

Phân loại các tasks (Type):

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

Phân loại độ ưu tiên (Priority)

1. **Low** : Độ ưu tiên của task thấp (Có thể hoàn thành quá hạn)
2. **Medium** : Độ ưu tiên của task trung bình (Hoàn thành đúng hạn)
3. **High** : Độ ưu tiên của task cao (Cần hoàn thành gấp)

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

Các Power-Ups khuyên dùng

- **! Amazing Fields !**
 - Tạo progress field dành cho những task lớn (những task này related với 1 board)
- **Card Dependencies by Screenful**
 - Tạo mối liên hệ giữa các card
- **Card Size by Screenful**
 - Hỗ trợ Story Points
 - Thống kê Story Points
- **Approvals for Trello**
 - Approval cho việc testing
- **GitHub**
 - Tích hợp với Github

Cách tính story point

Dựa vào thời gian tiêu chuẩn để làm xong 1 task, thời gian tương ứng với story point như sau:

Story Point	Time spent (hours)
-------------	--------------------

Story Point	Time spent (hours)
1 sp	0 - 0.5
2 sp	0.5 - 1.5
3 sp	1.5 - 3
5 sp	3 - 5.5
8 sp	5.5 - 9.5