

# 5. Git Workflow & Pull Request

## 5.1. Mục tiêu chương

Chương này quy định chuẩn commit và quy trình merge code nhằm:

- Đảm bảo lịch sử thay đổi rõ ràng, dễ truy vết
- Giảm xung đột giữa các nhóm phát triển
- Đảm bảo chất lượng code trước khi đưa vào nhánh chính
- Chuẩn hóa quy trình phát triển giữa Core Team và Partner

## 5.2. Khái niệm / phạm vi áp dụng

Quy định này áp dụng cho:

- Tất cả repository của dự án
- Core Team và Partner Team
- Mọi commit và Pull Request (PR)

Quy trình phát triển tuân theo:

- Chuẩn format commit
- Chiến lược branch

**Điều kiện merge bắt buộc**

## 5.3. Quy định chính

### 5.3.1. Format commit

Tất cả commit phải theo format:

<type>: <description>

Trong đó:

- <type>: loại thay đổi
- <description>: mô tả ngắn gọn nội dung thay đổi

### Ví dụ commit hợp lệ

feat: add planning API

fix: validate file upload

refactor: optimize planning service

docs: update API guideline

test: add unit test for planning service

chore: update dependency versions

---

### Type hợp lệ

Type	Ý nghĩa
feat	Thêm chức năng mới
fix	Sửa lỗi
refactor	Tối ưu, không đổi logic
docs	Cập nhật tài liệu
test	Thêm hoặc sửa test
chore	Thay đổi cấu hình, build, dependency

---

## 5.3.2. Branch strategy

Hệ thống sử dụng mô hình branch chuẩn:

Branch	Mục đích
main	Bản stable, dùng cho production
develop	Bản tích hợp, nơi merge các feature
feature/*	Phát triển chức năng mới
fix/*	Sửa lỗi

---

## Quy tắc làm việc với branch

- Không commit trực tiếp vào main
  - Không commit trực tiếp vào develop
  - Mọi thay đổi phải qua Pull Request
  - Branch phải được đặt tên đúng quy tắc
- 

## Ví dụ tên branch hợp lệ

feature/planning-create-api

feature/document-upload

fix/login-null-pointer

fix/planning-date-validation

---

## 5.3.3. Điều kiện merge Pull Request

Pull Request chỉ được merge khi đáp ứng đầy đủ các điều kiện:

1. Build thành công

Không sửa:

common/\*\*

config/\*\*

2. Không sửa shared entity

**Có review từ Core Team**

## 5.4. Cách thực hiện / quy trình

### Quy trình phát triển tính năng

Bước 1: Tạo branch từ develop

```
git checkout develop
```

```
git pull
```

```
git checkout -b feature/planning-create-api
```

Bước 2: Thực hiện commit theo chuẩn

```
git commit -m "feat: add planning create API"
```

Bước 3: Push branch lên remote

```
git push origin feature/planning-create-api
```

Bước 4: Tạo Pull Request vào develop

Bước 5: Core Team review

Bước 6: Nếu đạt yêu cầu → merge vào develop

---

## Quy trình sửa lỗi

Bước 1: Tạo branch fix

git checkout develop

git checkout -b fix/planning-null-error

Bước 2: Commit

git commit -m "fix: handle null planning name"

Bước 3: Tạo PR và chờ review

## 5.5. Ví dụ minh họa

### Trường hợp commit sai

```
PS D:\Project\angular base\fe-app> git commit -n -m 'update'  
[fix/planning-null-error dc80f97] update  
1 file changed, 1 insertion(+), 1 deletion(-)  
PS D:\Project\angular base\fe-app> █
```

Sai vì:

- Không theo format chuẩn
  - Không có type
-

# Trường hợp commit đúng

fix: handle null planning name

```
PS D:\Project\angular base\fe-app> git checkout -b fix/planning-null-error
Switched to a new branch 'fix/planning-null-error'
PS D:\Project\angular base\fe-app> git commit -m 'fix: handle null planning name'
[fix/planning-null-error f47deb8] fix: handle null planning name
1 file changed, 1 insertion(+), 1 deletion(-)
PS D:\Project\angular base\fe-app> █
```

---

# Trường hợp merge sai

Partner:

- Commit trực tiếp vào develop
- Không tạo PR
- Không có review

→ Vi phạm quy trình.

---

# Trường hợp merge đúng

1. Tạo branch feature/planning-api
2. Commit theo chuẩn
3. Tạo PR vào develop
4. Core Team review
5. Build pass
6. Merge

```
● 1 file changed, 1 insertion(+), 1 deletion(-)
PS D:\Project\angular base\fe-app> git checkout -b feature/planning-api
Switched to a new branch 'feature/planning-api'
● PS D:\Project\angular base\fe-app> git checkout -b feature/planning-api-add
Switched to a new branch 'feature/planning-api-add'
● PS D:\Project\angular base\fe-app> git checkout feature/planning-api
M       src/app/features/theme/theme-config-popup.component.ts
● M       src/styles.scss
Switched to branch 'feature/planning-api'
PS D:\Project\angular base\fe-app> git pull origin feature/planning-api-add
●
```

## 5.6. Checklist áp dụng

Trước khi tạo Pull Request:

- Commit đúng format <type>: <description>
- Branch đúng quy tắc feature/\* hoặc fix/\*
- Không commit trực tiếp vào main hoặc develop
- Build thành công
- Không sửa:
  - common/\*\*
  - config/\*\*
  - shared entity
- PR đã được Core Team review

---

Phiên bản #3

Được tạo 2026-02-23 07:50:11 UTC bởi Admin

Được cập nhật 2026-02-24 10:20:21 UTC bởi Nam Đặng