

2. Quy chuẩn phát triển module

2.1. Mục tiêu chương

Chương này quy định cấu trúc chuẩn của mỗi module và trách nhiệm của từng layer nhằm:

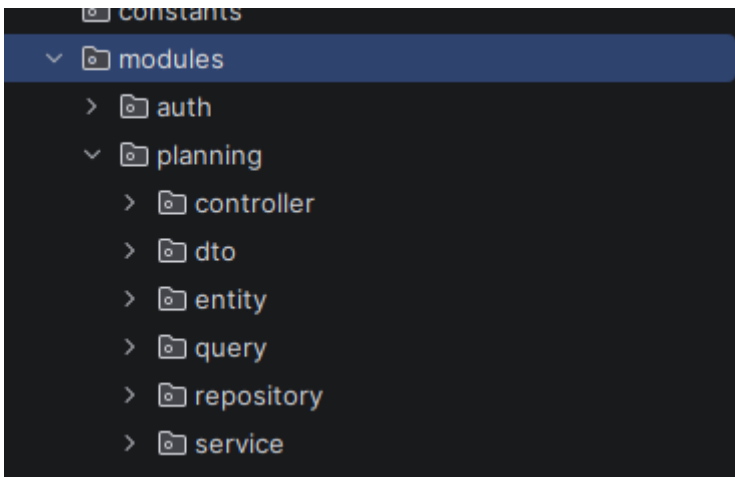
- Tách biệt rõ ràng các tầng xử lý
- Giảm phụ thuộc giữa các thành phần
- Dễ bảo trì, mở rộng và test
- Đảm bảo tính nhất quán giữa các module trong hệ thống

2.2. Khái niệm / phạm vi áp dụng

Quy tắc này áp dụng cho:

- Tất cả các module trong thư mục modules/**
- Cả Core Team và Partner Team
- Tất cả các chức năng mới được phát triển

Cấu trúc chuẩn mỗi module



modules/<module-name>/

controller/

service/

repository/

entity/

dto/

Ý nghĩa từng thành phần

Layer	Vai trò	ví dụ
controller	Nhận request từ client và trả response	<pre>@PostMapping public ApiResponse<PlanningResponse> create(@Valid @RequestBody PlanningCreateRequest req) { return ApiResponse.ok(service.create(req)); }</pre>
service	Xử lý logic nghiệp vụ	<pre>public PlanningResponse get(Long id) { PlanningEntity e = repo.findById(id).orElseThrow(() -> new AppException(ErrorCode.NOT_FOUND, HttpStatus.NOT_FOUND, "Planning not found: " + id)); return toResponse(e); }</pre>

repository	Truy cập và thao tác dữ liệu	<pre>public interface PlanQueryRepository { long count(PlanListFilter filter); List<PlanListItem> list(PlanListFilter filter); }</pre>
entity	Mapping bảng database	<pre>public class PlanningEntity { @Id @GeneratedValue(strategy = GenerationType.IDENTITY) private Long id; }</pre>
dto	Model request và response	<pre>public class PlanListFilter { private String keyword; private Integer page = 1; // 1-based private Integer size = 20; private String sortBy = "created_at"; private String sortDir = "desc"; public String getKeyword() { return keyword; } public void setKeyword(String keyword) { this.keyword = keyword; } public Integer getPage() { return page; } public void setPage(Integer page) { this.page = page; } public Integer getSize() { return size; } public void setSize(Integer size) { this.size = size; } public String getSortBy() { return sortBy; } public void setSortBy(String sortBy) { this.sortBy = sortBy; } public String getSortDir() { return sortDir; } public void setSortDir(String sortDir) { this.sortDir = sortDir; } }</pre>

2.3. Quy định chính

Trách nhiệm từng layer

Controller

- Nhận request từ client

- Gọi service tương ứng
- Trả response về client
- Không xử lý logic nghiệp vụ

Service

- Xử lý logic nghiệp vụ
- Điều phối repository
- Chuyển đổi entity ↔ DTO

Repository

- Truy cập database
- Thực hiện các thao tác CRUD
- Không chứa logic nghiệp vụ

Entity

- Mapping với bảng database
- Không chứa logic nghiệp vụ phức tạp

DTO

- Model cho request/response
- Không gắn trực tiếp với entity

Quy tắc bắt buộc

1. Controller không được chứa logic nghiệp vụ
2. Không trả trực tiếp Entity ra API
3. Luôn dùng DTO cho request/response
4. Response chuẩn toàn hệ thống: ApiResponse<T>

2.4. Cách thực hiện / quy trình

Quy trình phát triển một chức năng mới

Bước 1: Tạo DTO

Ví dụ:

UserCreateRequest

UserResponse

Bước 2: Tạo Entity

Mapping với bảng database:

UserEntity

Bước 3: Tạo Repository

UserRepository

Bước 4: Tạo Service

UserService

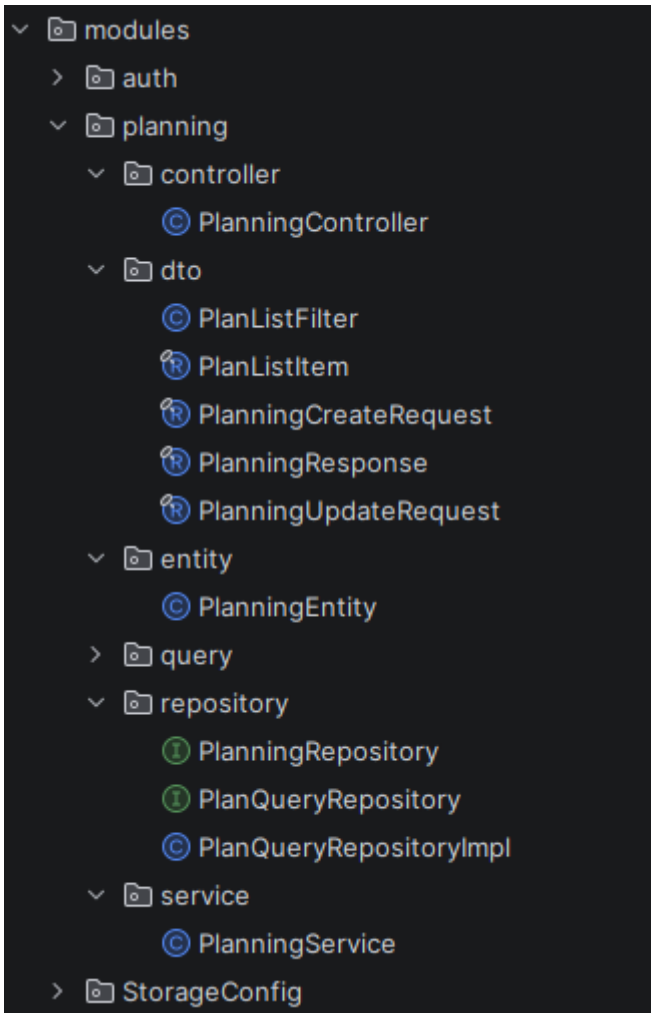
Xử lý logic:

- Validate dữ liệu
- Gọi repository
- Mapping entity → DTO

Bước 5: Tạo Controller

UserController

- Nhận request
- Gọi service



2.5. Ví dụ minh họa

Ví dụ sai (vi phạm quy tắc)

Controller chứa logic:

```
@GetMapping("/users")
```

```
public List<UserEntity> getAll() {
```

```
    return userRepository.findAll();
```

```
}
```

Sai vì:

- Trả Entity trực tiếp
- Không dùng Service

- Không dùng ApiResponse
-

Ví dụ đúng

```
@GetMapping("/users")
```

```
public ApiResponse<List<UserResponse>> getAll() {  
    return ApiResponse.ok(userService.getAll());  
}
```

Service:

```
public List<UserResponse> getAll() {  
    return userRepository.findAll()  
        .stream()  
        .map(this::toResponse)  
        .toList();  
}
```

2.6. Checklist áp dụng

Trước khi commit module mới:

- Có đủ:
 - controller/
 - service/
 - repository/
 - entity/
 - dto/

- Controller không chứa logic nghiệp vụ
- Không trả Entity ra API
- Tất cả API trả về ApiResponse<T>
- DTO tách biệt với Entity

Phiên bản #1

Được tạo 2026-02-23 10:22:20 UTC bởi admin_lifetex

Được cập nhật 2026-02-23 10:22:20 UTC bởi admin_lifetex