

Coding Standards

1. Nguyên tắc chung.

Codebase tuân thủ các tiêu chuẩn được sử dụng phổ biến trong môi trường chuyên nghiệp để đảm bảo tính bảo trì, tính dễ đọc và sự nhất quán. Một số tiêu chuẩn:

- Hạn chế sử dụng biến toàn cục, nó sẽ khó debug, testing.
- Naming conventions cho local variables, global variables, constants và functions:
Thống nhất các đặt biến và hàm.
- Indentation: Giúp code dễ đọc bằng cách sử dụng khoảng trắng chính xác.
- Quy ước cho xử lý ngoại lệ và lỗi trả về.

2. Áp dụng

a) Standard headers:

Trong backend, các file .service đều có phần mô tả chung (header) ở đầu file. Nội dung bao gồm: tên, tác giả, ngày tạo, ngày cập nhật, tóm tắt module, mô tả hàm và các biến toàn cục.

b) Naming Conventions.

Về cách đặt tên variable và function.

- Sử dụng tên gọi có ý nghĩa, dễ hiểu và chỉ rõ mục đích.
- Đặt tên có format camelCase: `mailAddress`, `handleClick()`
- Đặt tên có format PascalCase cho class và constructor: `BaseIdentityDocument`, `Address`
- Đặt tên có format UPPER_SNAKE_CASE cho hằng số (constant) `VITE_API_BASE_URL`, `STATUS_TRANSITIONS`
- Về cách đặt tên đường dẫn URL (route, endpoint): dùng `kebab-case`
 - `POST /class-registrations`
- Về cách đặt tên file và thư mục:
 - Sử dụng format camelCase: `multiSelect.tsx`, `usePrograms.ts`, `courseRoute.js`
 - Đối với server side, các file trong module quan trọng như controllers, routes, services sẽ được kèm hậu tố `<tên module>`. Ví dụ `class.service.js`, `program.controller.js`
 - Các file test sẽ chứa hậu tố `.test`, các file mock test sẽ chứa `.mock`. Chẳng hạn, `student.service.test.js`, `course.mock.js`

c) Code Structure and Formatting.

- Indentation và Spacing.
 - Sử dụng 2 dấu khoảng trắng cho indentation (không dùng tab).
 - Luôn sử dụng dấu chấm phẩy ở cuối dòng code.
- Comments.
 - Có inline-comment cho những đoạn code phức tạp.

d) Error Handling.

- Luôn handle lỗi rõ ràng, không bao giờ bỏ qua.
- Luôn sử dụng khối lệnh try-catch trong các function, service quan trọng.
- Trả về thông báo lỗi có ý nghĩa với context.
- Định dạng lỗi thống nhất: Nhóm định nghĩa sẵn một cấu trúc lỗi cơ bản (BaseError) bao gồm:
 - message: thông báo lỗi chính (tiếng Anh),
 - message_vi: thông báo lỗi tiếng Việt,
 - statusCode: mã trạng thái HTTP tương ứng.

Từ đó, xây dựng các loại lỗi kế thừa như:

- NotFoundError: lỗi không tìm thấy tài nguyên.
- ValidationError: lỗi do dữ liệu không hợp lệ.
- InternalServerError: lỗi hệ thống.
- DuplicateResourceError: lỗi khi tài nguyên đã tồn tại.