

**BỘ KHOA HỌC VÀ CÔNG NGHỆ**  
**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**



**Môn: THỰC TẬP CƠ SỞ**

**Báo cáo tuần**

**Website quản lý lớp học dành cho giáo viên và học sinh**

Giảng viên: Kim Ngọc Bách  
Học và tên: Nguyễn Tuấn Anh  
Mã sinh viên: B22DCVT026  
Lớp: E22CQCN05-B

*Hà Nội - 2025*

**Kính gửi Thầy,**

Em xin gửi báo cáo cá nhân tuần 6 cho dự án trang web quản lý lớp học. Trong tuần này, em đã tiếp tục phát triển backend, hoàn thiện các endpoint API RESTful cho quản lý lớp học và bài tập, tích hợp PHPMailer để gửi thông báo nộp bài, viết tài liệu API cơ bản và kiểm thử các endpoint. Báo cáo này trình bày chi tiết các công việc em đã thực hiện.

## **Mục tiêu**

- Xây dựng các endpoint API RESTful cho quản lý lớp học (tạo, tham gia, xem danh sách) và bài tập (giao bài, nộp bài, xem bài nộp).
- Tích hợp PHPMailer để gửi thông báo nộp bài.
- Viết tài liệu API cơ bản, bao gồm mô tả endpoint, tham số và định dạng phản hồi.
- Kiểm thử tất cả các endpoint bằng Postman để đảm bảo hoạt động chính xác.
- Tiếp tục áp dụng các biện pháp bảo mật.

## **Công việc Thực hiện**

### **1. Phát triển API RESTful (PHP Thuần)**

Em tiếp tục phát triển các endpoint API trong PHP thuần, bổ sung các chức năng sau:

- **POST /api/classes/create:**
  - **Chức năng:** Giáo viên tạo lớp học mới, lưu vào bảng classes.
  - **Tham số:** name, code, description, teacher\_id.
  - **Xử lý:**
    - Kiểm tra code duy nhất và teacher\_id hợp lệ.
    - Lưu thông tin vào bảng classes.
    - Trả về JSON: { "success": true, "class\_id": 123 }.
- **POST /api/classes/join:**
  - **Chức năng:** Học sinh tham gia lớp học bằng mã lớp, lưu vào bảng class\_enrollments.
  - **Tham số:** code, student\_id.
  - **Xử lý:**
    - Kiểm tra mã lớp tồn tại và học sinh chưa tham gia.
    - Thêm bản ghi vào class\_enrollments.
    - Trả về JSON: { "success": true, "message": "Tham gia thành công" }.

- **GET /api/classes/list:**
  - **Chức năng:** Lấy danh sách lớp học của người dùng (giáo viên hoặc học sinh).
  - **Tham số:** user\_id, role.
  - **Xử lý:**
    - Nếu là giáo viên, lấy các lớp từ classes với teacher\_id.
    - Nếu là học sinh, lấy các lớp từ class\_enrollments.
    - Trả về JSON: { "success": true, "classes": [{ "id": 123, "name": "Lớp Toán", "code": "MATH001" }, ...] }.
- **POST /api/assignments/create:**
  - **Chức năng:** Giáo viên tạo bài tập mới, lưu vào bảng assignments.
  - **Tham số:** title, description, deadline, class\_id.
  - **Xử lý:**
    - Kiểm tra class\_id hợp lệ và người dùng là giáo viên.
    - Lưu bài tập vào bảng assignments.
    - Trả về JSON: { "success": true, "assignment\_id": 456 }.
- **POST /api/submissions/submit:**
  - **Chức năng:** Học sinh nộp bài tập, lưu vào bảng submissions.
  - **Tham số:** student\_id, assignment\_id, file (tệp tải lên).
  - **Xử lý:**
    - Kiểm tra assignment\_id hợp lệ và học sinh đã tham gia lớp.
    - Lưu tệp vào thư mục uploads/ và đường dẫn vào file\_path.
    - Trả về JSON: { "success": true, "message": "Nộp bài thành công" }.
- **GET /api/submissions/list:**
  - **Chức năng:** Lấy danh sách bài nộp cho một bài tập (giáo viên) hoặc của một học sinh.
  - **Tham số:** assignment\_id (giáo viên) hoặc student\_id (học sinh).
  - **Xử lý:**
    - Truy vấn bảng submissions dựa trên tham số.
    - Trả về JSON: { "success": true, "submissions": [{ "id": 789, "student\_id": 101, "file\_path": "uploads/file.pdf" }, ...] }.

## 2. Tích hợp PHPMailer cho Thông báo Nộp Bài

- Em mở rộng tích hợp PHPMailer để gửi thông báo khi học sinh nộp bài:
  - **Chức năng:** Gửi email đến giáo viên khi có bài nộp mới, bao gồm tên học sinh, tiêu đề bài tập và thời gian nộp.
  - **Cấu hình:** Tiếp tục sử dụng Gmail SMTP từ Tuần 5.
  - **Thực hiện:** Trong endpoint /api/submissions/submit, em thêm đoạn mã gọi PHPMailer để gửi email đến giáo viên của lớp liên quan.

- **Kiểm tra:** Em thử nộp bài bằng Postman và xác nhận email được gửi thành công với nội dung chính xác.

### 3. Tài liệu API

Em đã viết tài liệu API cơ bản dưới dạng Markdown, bao gồm:

- **Mô tả Endpoint:**
  - Ví dụ: POST /api/register: Đăng ký người dùng mới.
    - Tham số: full\_name (string), email (string), password (string), role (enum: teacher, student).
    - Phản hồi: { "success": true, "message": "Đăng ký thành công" }.
- **Danh sách Endpoint:** Liệt kê tất cả 6 endpoint với tham số, định dạng phản hồi và mã lỗi (ví dụ: 400 cho dữ liệu không hợp lệ).
- **Hướng dẫn Sử dụng:** Cách gọi API bằng Postman với các ví dụ curl.
- Em lưu tài liệu trong tệp api-docs.md.

### 4. Kiểm thử API

- Em sử dụng Postman để kiểm tra tất cả 6 endpoint:
  - **Kịch bản thành công:** Tất cả endpoint trả về JSON đúng định dạng với mã trạng thái 200.
  - **Kịch bản lỗi:** Kiểm tra các trường hợp như email trùng lặp, mã lớp không tồn tại, tệp tải lên không hợp lệ. Mỗi trường hợp trả về JSON với mã lỗi (400, 401, 404) và thông báo chi tiết.
- Em phát hiện lỗi trong endpoint /api/submissions/submit khi tệp vượt quá kích thước cho phép, đã khắc phục bằng cách kiểm tra kích thước trước khi lưu.

### 5. Bảo mật

- **Mã hóa Mật khẩu:** Tiếp tục sử dụng bcrypt cho các endpoint liên quan đến người dùng.
- **Kiểm tra Dữ liệu:** Em áp dụng filter\_var() và preg\_match() để kiểm tra định dạng mã lớp và thời hạn bài tập.
- **CSRF:** Em thêm kiểm tra token CSRF trong các endpoint POST bằng cách so sánh token trong header với token lưu trong session.
- **Tệp Tải lên:** Em giới hạn định dạng tệp (pdf, docx) và kích thước tối đa (5MB) trong endpoint /api/submissions/submit.

## Kết quả

- **API Endpoint:** Hoàn thành 6 endpoint (/api/register, /api/login, /api/classes/create, /api/classes/join, /api/classes/list, /api/assignments/create, /api/submissions/submit, /api/submissions/list).
- **PHPMailer:** Tích hợp thông báo nộp bài thành công.
- **Tài liệu API:** Tập api-docs.md với mô tả chi tiết các endpoint.
- **Kiểm thử:** Tất cả endpoint hoạt động đúng qua Postman.
- **Bảo mật:** Áp dụng mã hóa, kiểm tra dữ liệu và CSRF cơ bản.

## Thách thức và Giải pháp

- **Thách thức:** Xử lý tải tệp trong endpoint /api/submissions/submit gặp lỗi khi thư mục uploads/ chưa tồn tại.
  - **Giải pháp:** Em thêm mã kiểm tra và tự động tạo thư mục nếu cần.
- **Thách thức:** Viết tài liệu API rõ ràng và đầy đủ trong thời gian ngắn.
  - **Giải pháp:** Em tham khảo mẫu tài liệu từ các dự án mã nguồn mở để chuẩn hóa định dạng.

## Kế hoạch Tiếp theo

- Tích hợp frontend (Tuần 3-4) với backend qua các endpoint API.
- Thêm các endpoint phụ như chỉnh sửa/xóa lớp học, chấm điểm bài nộp.
- Thực hiện kiểm tra toàn diện hệ thống (frontend + backend).