

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 5 cho dự án trang web quản lý lớp học. Trong tuần này, em đã tập trung vào việc cài đặt môi trường phát triển, xây dựng cơ sở dữ liệu MySQL, bắt đầu phát triển API RESTful bằng PHP thuần, và tích hợp PHPMailer để gửi email. Báo cáo này trình bày chi tiết các công việc em đã thực hiện một mình để đảm bảo tiến độ dự án.

Mục tiêu

- Cài đặt môi trường phát triển cục bộ bằng XAMPP, cấu hình PHP và MySQL.
- Tạo cơ sở dữ liệu MySQL dựa trên thiết kế ERD từ Tuần 2, đảm bảo tối ưu hóa và các mối quan hệ chính xác.
- Xây dựng một số endpoint API RESTful cho các chức năng chính (đăng ký, đăng nhập).
- Tích hợp PHPMailer để gửi email xác thực tài khoản.
- Áp dụng các biện pháp bảo mật cơ bản như mã hóa mật khẩu và kiểm tra dữ liệu đầu vào.

Công việc Thực hiện

1. Cài đặt Môi trường Phát triển

- **XAMPP:** Em đã cài đặt XAMPP trên máy tính cá nhân để thiết lập môi trường cục bộ với Apache, PHP 8.2, và MySQL. Em kiểm tra cấu hình để đảm bảo PHP và MySQL hoạt động ổn định.
- **Cấu hình:** Em điều chỉnh tệp php.ini để tăng giới hạn kích thước tải lên tệp (upload_max_filesize = 10M) nhằm hỗ trợ tải bài nộp. Em cũng bật module openssl và mbstring để hỗ trợ mã hóa và xử lý chuỗi.
- **Kiểm tra:** Em chạy một trang PHP thử nghiệm (info.php) để xác nhận môi trường hoạt động đúng, hiển thị thông tin PHP và kết nối MySQL thành công.

2. Tạo Cơ sở Dữ liệu MySQL

Dựa trên ERD từ Tuần 2, em đã tạo cơ sở dữ liệu classroom_management với các bảng sau:

- **users:**
 - id: INT, Primary Key, Auto-increment
 - full_name: VARCHAR(100)

- email: VARCHAR(100), Unique
 - password: VARCHAR(255), Hashed (bcrypt)
 - role: ENUM('teacher', 'student', 'admin')
 - created_at: DATETIME
 - Index: email
- **classes:**
 - id: INT, Primary Key, Auto-increment
 - name: VARCHAR(100)
 - code: VARCHAR(10), Unique
 - teacher_id: INT, Foreign Key (references users.id)
 - description: TEXT
 - Index: code
- **class_enrollments:**
 - student_id: INT, Foreign Key (references users.id)
 - class_id: INT, Foreign Key (references classes.id)
 - enrolled_at: DATETIME
 - Primary Key: (student_id, class_id)
- **assignments:**
 - id: INT, Primary Key, Auto-increment
 - title: VARCHAR(100)
 - description: TEXT
 - deadline: DATETIME
 - class_id: INT, Foreign Key (references classes.id)
- **submissions:**
 - id: INT, Primary Key, Auto-increment
 - student_id: INT, Foreign Key (references users.id)
 - assignment_id: INT, Foreign Key (references assignments.id)
 - file_path: VARCHAR(255)
 - submitted_at: DATETIME
 - grade: DECIMAL(5,2), Nullable
 - feedback: TEXT, Nullable
- **Tối ưu hóa:** Em thêm các chỉ mục trên email và code để tăng tốc truy vấn. Em sử dụng InnoDB để hỗ trợ khóa ngoại và giao dịch.
- **Dữ liệu mẫu:** Em chèn dữ liệu thử nghiệm (3 giáo viên, 10 học sinh, 5 lớp học, 10 bài tập) để kiểm tra các mối quan hệ và ràng buộc.

3. Phát triển API RESTful (PHP Thuần)

Em quyết định sử dụng PHP thuần để xây dựng API RESTful, thay vì Laravel, để đơn giản hóa và kiểm soát tốt hơn trong phạm vi dự án cá nhân. Em đã phát triển các endpoint sau:

- **POST /api/register:**

- **Chức năng:** Xử lý đăng ký người dùng, lưu thông tin vào bảng users và gửi email xác nhận.
- **Tham số:** full_name, email, password, role.
- **Xử lý:**
 - Kiểm tra định dạng email và độ dài mật khẩu (tối thiểu 8 ký tự).
 - Mã hóa mật khẩu bằng password_hash() với bcrypt.
 - Lưu thông tin vào bảng users.
 - Trả về JSON: { "success": true, "message": "Đăng ký thành công" }.
- **POST /api/login:**
 - **Chức năng:** Xác thực thông tin đăng nhập và tạo token phiên.
 - **Tham số:** email, password.
 - **Xử lý:**
 - Kiểm tra email tồn tại trong bảng users.
 - Xác minh mật khẩu bằng password_verify().
 - Tạo token phiên đơn giản (chuỗi ngẫu nhiên) và lưu vào session.
 - Trả về JSON: { "success": true, "token": "xyz", "role": "student" }.
- **Cấu trúc API:** Em tổ chức mã nguồn trong thư mục api/, với các tệp như register.php và login.php. Mỗi endpoint xử lý dữ liệu đầu vào qua \$_POST và trả về dữ liệu JSON.

4. Tích hợp PHPMailer

- **Cài đặt:** Em tích hợp PHPMailer qua Composer để gửi email xác nhận tài khoản.
- **Cấu hình:** Sử dụng Gmail SMTP với thông tin xác thực (smtp.gmail.com, port 587, TLS). Em tạo một tài khoản Gmail thử nghiệm để gửi email.
- **Chức năng:** Khi người dùng đăng ký, PHPMailer gửi email chứa liên kết xác nhận (giả lập với mã tĩnh). Nội dung email bao gồm tên người dùng và hướng dẫn xác nhận.
- **Kiểm tra:** Em thử gửi email đến một địa chỉ cá nhân và xác nhận email được gửi thành công.

5. Bảo mật

- **Mã hóa Mật khẩu:** Sử dụng password_hash() với bcrypt cho mật khẩu.
- **Kiểm tra Dữ liệu:** Em áp dụng filter_var() để kiểm tra định dạng email và htmlspecialchars() để ngăn chặn XSS trong dữ liệu đầu vào.

- **CSRF:** Em thêm một token CSRF đơn giản trong các biểu mẫu (dự kiến tích hợp với frontend sau).

6. Kiểm thử

- Em sử dụng Postman để kiểm tra các endpoint `/api/register` và `/api/login`.
Kết quả:
 - Đăng ký: Trả về JSON thành công khi dữ liệu hợp lệ, báo lỗi khi email trùng lặp.
 - Đăng nhập: Trả về token khi thông tin đúng, báo lỗi khi sai email/mật khẩu.
- Em phát hiện lỗi nhỏ khi email xác nhận không gửi được do cấu hình SMTP sai, đã khắc phục bằng cách cập nhật thông tin xác thực.

Kết quả

- **Môi trường Phát triển:** XAMPP được cài đặt và cấu hình thành công.
- **Cơ sở Dữ liệu:** Cơ sở dữ liệu `classroom_management` với 5 bảng, tối ưu hóa và dữ liệu mẫu.
- **API Endpoint:** Hoàn thành 2 endpoint (`/api/register`, `/api/login`) với mã nguồn PHP thuần.
- **PHPMailer:** Tích hợp thành công, gửi được email xác nhận.
- **Bảo mật:** Áp dụng mã hóa mật khẩu và kiểm tra dữ liệu cơ bản.

Thách thức và Giải pháp

- **Thách thức:** Cấu hình PHPMailer với Gmail SMTP gặp lỗi xác thực.
 - **Giải pháp:** Em bật “Less secure app access” trong tài khoản Gmail thử nghiệm và kiểm tra lại thông tin SMTP.
- **Thách thức:** Xây dựng API RESTful bằng PHP thuần mất thời gian để xử lý JSON và lỗi.
 - **Giải pháp:** Em tham khảo tài liệu REST API và sử dụng `json_encode()` để chuẩn hóa đầu ra.

Kế hoạch Tuần 6

- Tiếp tục phát triển các endpoint API 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 mới bằng Postman.