PHỤ LỤC BÁO CÁO CÁ NHÂN

Họ tên: Vũ Hồng Nhung

MSSV: 20221864

Nhóm: group08

Tên đề tài: Diễn đàn hỏi đáp và chia sẻ tài liệu học tập

## 1. Lý do lựa chọn middleware trong một trường hợp cụ thể (1 điểm)

### 1.1. Miêu tả trường hợp

- Chức năng cụ thể cần xử lý: Tải lên tài liệu (file upload) cho chức năng đăng bài

- Tình huống cụ thể:

Trong dự án hệ thống chia sẻ tài liệu BKDocs, nhóm sử dụng middleware multer cho chức năng **tải lên tài liệu (file upload)**. Middleware này được chọn vì Node.js không hỗ trợ xử lý multipart/form-data (loại dữ liệu được gửi khi upload file) một cách tự động. Do đó, multer giúp tách và xử lý các phần dữ liệu là tệp đính kèm từ client.

**1.2. Lý do chọn middleware / thư viện bên thứ 3**

* multer là middleware phổ biến, đáng tin cậy, hỗ trợ cấu hình chi tiết về nơi lưu file, đổi tên file, giới hạn dung lượng,...
* Có cộng đồng lớn, tài liệu rõ ràng, dễ tích hợp vào Express.
* So với các thư viện như formidable, busboy thì multer đơn giản hơn để tích hợp nhanh cho nhu cầu cơ bản đến trung bình.

**1.3. Middleware được chọn**

**- Tên:** multer

- **Vai trò:** Middleware xử lý multipart/form-data để lấy file từ request gửi từ client.

- **Nhiệm vụ:**

* Đọc dữ liệu từ form.
* Lưu file vào thư mục chỉ định trên server.
* Trả về thông tin file để backend sử dụng (ví dụ: tên file, kích thước, đường dẫn,...)

**- Lý do chọn:** Dễ dùng, cấu hình rõ ràng, tương thích với Express, hiệu quả cho bài toán upload file kèm bài viết.

## 2. Giải thích đường đi từ request đến response (1 điểm)

**- Gửi từ form trên giao diện:**  
Người dùng nhập tiêu đề, nội dung và chọn file đính kèm → nhấn "Đăng bài" → fetch() gửi POST request với FormData đến /api/posts.

**- Qua route:**  
Route Express nhận request:  
app.post('/api/posts', upload.single('file'), postController.createPost)

**- Middleware xử lý:**  
multer xử lý file upload → lưu vào thư mục uploads/ → đính kèm thông tin vào req.file.

**- Controller xử lý logic:**  
Đọc dữ liệu từ req.body và req.file → kiểm tra hợp lệ → chuẩn bị dữ liệu lưu CSDL.

**- Truy cập CSDL nếu cần:**  
Lưu thông tin bài viết và đường dẫn file vào bảng posts trong MySQL.

**- Controller trả kết quả:**  
Trả về JSON: { success: true, message: "Đăng thành công" }

**- Render view (hoặc API trả JSON):**  
API trả JSON về cho client.

**- Client nhận và hiển thị:**  
Giao diện hiển thị thông báo thành công và cập nhật danh sách bài viết.

## 3. Giải thích vận hành website mình làm, tập trung vào phần việc được phân (1 điểm)

Phần việc mình thực hiện bao gồm:  
- Chức năng: ..............................................................  
- Mô tả chi tiết cách phần đó hoạt động, tương tác với các thành phần khác:  
 + Frontend sử dụng gì  
 + Backend xử lý gì  
 + CSDL truy vấn ra sao