

# YÊU CẦU CHUNG ĐỀ TÀI THỰC HÀNH PHÁT TRIỂN WEB

## 1. Yêu cầu về Kiến trúc & Công nghệ (Technical Requirements)

- **Mô hình hệ thống:** Bắt buộc tách biệt **Frontend** và **Backend** (Client-Server Architecture).
  - **Frontend:** Xây dựng theo hướng SPA (Single Page Application) sử dụng Framework/Library hiện đại như *ReactJS*, *VueJS*, *Angular*. Tư duy lập trình theo Component.
  - **Backend:** Xây dựng RESTful API chuẩn (trả về JSON). Có thể sử dụng *NodeJS (Express/NestJS)*, *PHP (Laravel)*, *Java (Spring Boot)*, *C# (.NET Core)*.
- **Cơ sở dữ liệu:**
  - Sử dụng Hệ quản trị CSDL quan hệ (*MySQL*, *SQL Server*, *PostgreSQL*).
  - **Bắt buộc:** Sử dụng kỹ thuật **Migrations** để quản lý phiên bản CSDL (không gửi file .sql xuất thủ công).
  - **Bắt buộc:** Có **Seeder/Factory** để sinh dữ liệu giả lập (tối thiểu 500-1000 bản ghi chính) phục vụ kiểm thử hiệu năng và phân trang.
- **Triển khai (Deployment):**
  - Mã nguồn phải được đóng gói bằng **Docker**.
  - Cung cấp file **docker-compose.yml** để giảng viên/hội đồng có thể chạy toàn bộ dự án (App + DB) chỉ bằng 1 câu lệnh.

## 2. Yêu cầu về Chức năng (Functional Requirements)

Sản phẩm phải giải quyết trọn vẹn bài toán nghiệp vụ, bao gồm các nhóm chức năng sau:

- Nhóm quản trị hệ thống (System Admin):
  - Đăng nhập/Đăng xuất/Quên mật khẩu (Gửi email reset).
  - Phân quyền (RBAC): Hệ thống phải có tối thiểu 3 vai trò (Roles) (VD: *Admin*, *Staff*, *Customer*). Admin có thể phân quyền truy cập cho Staff.
- Nhóm nghiệp vụ chính (Core Business):
  - Thực hiện trọn vẹn các thao tác CRUD (Thêm, Xem, Sửa, Xóa/Ẩn) cho các đối tượng chính.
  - Logic phức tạp: Phải có các chức năng xử lý logic nghiệp vụ, không chỉ nhập-xuất đơn thuần (VD: *Tính toán giá hàng*, *Kiểm tra tồn kho*, *Xử lý đặt lịch trùng giờ*, *Tính lương/thưởng...*).
  - Tìm kiếm & Lọc: Chức năng tìm kiếm đa tiêu chí, sắp xếp và phân trang (Pagination) bắt buộc.
- Nhóm báo cáo & Thống kê (Dashboard):
  - Tổng hợp số liệu (VD: *Tổng doanh thu tháng*, *Số lượng đơn hàng mới*).
  - Biểu đồ (Chart): Tối thiểu 2 biểu đồ trực quan (Cột, Tròn, Đường...) sử dụng thư viện như *ChartJS*, *Recharts*.
- Tính năng nâng cao (Điểm cộng):
  - Real-time: Thông báo tức thời (Notification) hoặc Chat trực tuyến (sử dụng *Socket.io/SignalR*).
  - Export/Import: Xuất dữ liệu ra Excel/PDF hoặc Nhập liệu từ Excel.

### 3. Yêu cầu về Giao diện & Trải nghiệm (UI/UX Requirements)

- Responsive Design: Giao diện hiển thị tốt trên cả Máy tính (Desktop) và Thiết bị di động (Mobile/Tablet).
- Validation (Kiểm tra dữ liệu):
  - Bắt lỗi nhập liệu ngay tại Frontend (VD: Email sai định dạng, Bỏ trống trường bắt buộc).
  - Hiển thị thông báo (Toast/Alert) rõ ràng khi thao tác thành công hoặc thất bại.
- Thẩm mỹ: Giao diện hiện đại, bố cục rõ ràng, thống nhất về màu sắc và font chữ. Khuyến khích sử dụng UI Library (Ant Design, Material UI, TailwindCSS, Bootstrap 5).

### 4. Yêu cầu về Bảo mật (Security Requirements)

- Xác thực (Authentication): Sử dụng cơ chế JWT (JSON Web Token). Không lưu mật khẩu dạng rõ (Plain text) trong Database (phải Hash bằng Bcrypt/Argon2).
- An toàn dữ liệu:
  - Chống SQL Injection (Sử dụng ORM hoặc Prepared Statements).
  - Chống XSS (Cross-site Scripting) khi hiển thị dữ liệu người dùng nhập.
  - Validation chặt chẽ tại Backend (không chỉ tin tưởng Frontend).

### 5. Yêu cầu về Quy trình & Mã nguồn (Process & Code Quality)

- Quản lý mã nguồn: Sử dụng Git và lưu trữ trên GitHub/GitLab.
  - Lịch sử commit rõ ràng, không commit 1 lần toàn bộ dự án.
  - Áp dụng mô hình nhánh cơ bản (Main/Dev).
- Chất lượng mã nguồn:
  - Tuân thủ Coding Convention (quy tắc đặt tên biến, hàm).
  - Code được tổ chức rõ ràng theo mô hình MVC hoặc cấu trúc của Framework.
  - Không để lộ các thông tin nhạy cảm (API Key, DB Password) trong code (Sử dụng file .env).

### 6. Sản phẩm bàn giao (Deliverables)

Kết thúc 330 giờ làm việc, sinh viên cần nộp:

1. **Mã nguồn (Source Code):** Link Repository GitHub/GitLab (đã bao gồm file cấu hình Docker).
2. **Cơ sở dữ liệu:** File Migrations và Seeder (hoặc Script SQL dự phòng).
3. **Báo cáo thuyết minh (Technical Report):**
  - Mô tả bài toán, Biểu đồ Use Case, ERD, Kiến trúc hệ thống.
  - Tài liệu đặc tả API (Link Swagger hoặc file xuất từ Postman).
  - Hình ảnh giao diện và hướng dẫn sử dụng.
4. **Video Demo:** Clip ngắn (5-7 phút) quay lại quy trình hoạt động của các chức năng chính.
5. **Slide thuyết trình:** Dùng cho buổi bảo vệ cuối kỳ.