

THIẾT KẾ KIẾN TRÚC HỆ THỐNG

1. Lựa chọn mô hình kiến trúc

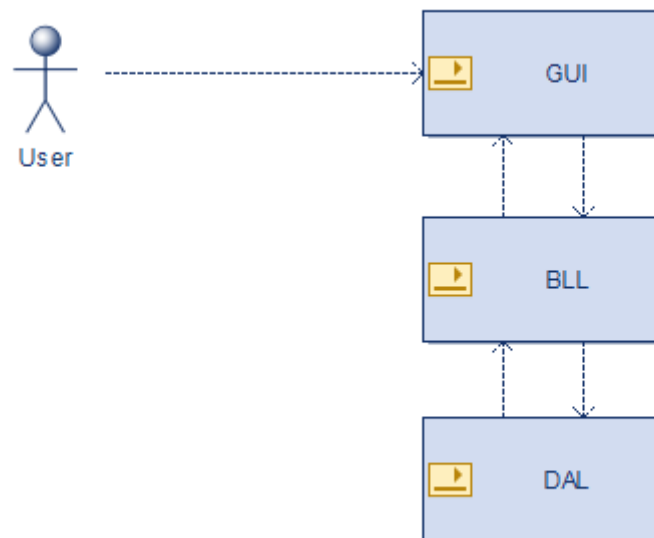
Để đảm bảo tính tổ chức, dễ bảo trì và mở rộng, dự án "Website Bán hàng E-commerce VM-Style" lựa chọn áp dụng mô hình **Kiến trúc 3 Lớp (3-Layer Architecture)**.

Mô hình này chia ứng dụng thành ba lớp (layer) logic riêng biệt:

1. **Presentation Layer (Tầng Giao diện):** Chịu trách nhiệm hiển thị thông tin, giao tiếp với người dùng và tiếp nhận yêu cầu.
2. **Business Logic Layer (BLL - Tầng Nghiệp vụ):** Chứa các quy tắc nghiệp vụ, logic xử lý, tính toán và điều phối luồng dữ liệu. Đây là "bộ não" của hệ thống.
3. **Data Access Layer (DAL - Tầng Truy cập Dữ liệu):** Chịu trách nhiệm giao tiếp trực tiếp với Cơ sở dữ liệu (CSDL) như đọc, ghi, cập nhật, xóa dữ liệu.

2. Sơ đồ kiến trúc tổng quan

Dưới đây là sơ đồ kiến trúc tổng quan của hệ thống dựa trên mô hình 3 lớp.



Mô tả sơ đồ:

- **Người dùng (Khách hàng / Admin)** tương tác với hệ thống thông qua Trình duyệt web (Client).
- Mọi yêu cầu (Request) từ Client sẽ đi đến **Presentation Layer(GUI)**.
- **Presentation Layer** gọi các hàm xử lý tại **Business Logic Layer (BLL)**.
- **BLL** thực thi các nghiệp vụ, nếu cần dữ liệu, BLL sẽ yêu cầu **Data Access Layer (DAL)**.
- **DAL** thực thi các truy vấn (SQL) đến **Cơ sở dữ liệu (Database)**.
- Dữ liệu (Data) được trả về theo luồng ngược lại: Database -> DAL -> BLL -> Presentation -> Client (Response).
- **Quan trọng:** Một lớp chỉ được phép giao tiếp trực tiếp với lớp ngay bên dưới nó (ví dụ: Presentation không được phép gọi trực tiếp DAL).

3. Mô tả luồng tương tác của Module

Để làm rõ cách các module tương tác, chúng ta xét luồng xử lý cho chức năng "Xác nhận Đặt hàng" (UC-02) qua 3 lớp:

Bối cảnh: Khách hàng đã ở trang Thanh toán và nhấn nút "Xác nhận Đặt hàng".

3.1. Tầng Giao diện (Presentation Layer - GUI)

- **Hành động:** Trình duyệt của Khách hàng gửi một HTTP Request (ví dụ: POST /api/orders) chứa thông tin (Họ tên, SĐT, Địa chỉ) và thông tin giỏ hàng đến máy chủ.
- **Module:** OrderController (một thành phần trong Presentation Layer) tiếp nhận yêu cầu này.
- **Xử lý:** OrderController xác thực dữ liệu đầu vào (đảm bảo không rỗng). Sau đó, nó gọi dịch vụ nghiệp vụ tương ứng là OrderService (thuộc tầng BLL) và truyền dữ liệu xuống.

3.2. Tầng Nghiệp vụ (Business Logic Layer - BLL)

- **Module:** OrderService (Dịch vụ Đơn hàng) được kích hoạt.
- **Xử lý (Logic chính):**
 1. OrderService gọi ProductService (một dịch vụ khác trong BLL) để kiểm tra tính hợp lệ của giỏ hàng (ví dụ: kiểm tra tồn kho).
 2. ProductService yêu cầu ProductRepository (thuộc DAL) lấy thông tin tồn kho mới nhất.
 3. Nếu tồn kho hợp lệ, OrderService thực hiện các nghiệp vụ: tính tổng tiền, phí vận chuyển, v.v.

4. OrderService gọi OrderRepository (thuộc DAL) và yêu cầu "Tạo đơn hàng mới" với các thông tin đã xử lý.
5. OrderService gọi ProductRepository (thuộc DAL) và yêu cầu "Cập nhật lại số lượng tồn kho" cho các sản phẩm đã bán.
6. OrderService gọi EmailService để gửi mail xác nhận cho khách hàng.
7. OrderService trả kết quả (thành công/thất bại) về cho OrderController.

3.3. Tầng Truy cập Dữ liệu (Data Access Layer - DAL)

- **Module:** OrderRepository và ProductRepository (Các kho lưu trữ dữ liệu).
- **Xử lý:**
 - **Khi OrderRepository được gọi (bởi BLL):** Nó biên dịch yêu cầu thành các câu lệnh SQL (hoặc ORM) để INSERT dữ liệu vào bảng Orders và OrderItems trong CSDL.
 - **Khi ProductRepository được gọi (bởi BLL):** Nó biên dịch yêu cầu thành câu lệnh SQL (hoặc ORM) để UPDATE (trừ) số lượng tồn kho (quantity) trong bảng Products.

3.4. Luồng phản hồi (Response)

- **DAL** báo cáo kết quả (ví dụ: thêm thành công) về cho **BLL**.
- **BLL** tổng hợp kết quả và báo cáo về cho **Presentation Layer**.
- **Presentation Layer** (OrderController) tạo một HTTP Response (ví dụ: mã 200 OK kèm mã đơn hàng) và gửi trả về cho Trình duyệt (Client).
- Trình duyệt hiển thị trang "Đặt hàng thành công" cho Khách hàng.