Mô hình MVC (Model-View-Controller)

**Phân tách trách nhiệm (Separation of Concerns)**

**Model**: Đại diện cho dữ liệu và logic nghiệp vụ. Nó quản lý cấu trúc dữ liệu, lưu trữ, truy xuất và xử lý logic cốt lõi của ứng dụng (ví dụ: truy vấn cơ sở dữ liệu, tính toán).

**View**: Chịu trách nhiệm hiển thị dữ liệu cho người dùng. Đây là giao diện người dùng (UI), đảm bảo dữ liệu từ Model được trình bày một cách trực quan (ví dụ: HTML, giao diện đồ họa).

**Controller**: Kết nối Model và View. Nó xử lý các yêu cầu từ người dùng, tương tác với Model để lấy hoặc cập nhật dữ liệu, sau đó gửi dữ liệu đến View để hiển thị.

**Tính mô-đun (Modularity)**

Mỗi thành phần (Model, View, Controller) hoạt động độc lập, giúp dễ dàng phát triển, bảo trì và mở rộng ứng dụng.

**Khả năng tái sử dụng (Reusability)**

Một Model có thể được sử dụng cho nhiều View khác nhau.

Ví dụ: Dữ liệu từ một Model có thể được hiển thị trên giao diện web hoặc ứng dụng di động mà không cần thay đổi Model.

**Dễ kiểm thử (Testability)**

Do các thành phần tách biệt, việc viết unit test cho từng phần (Model, Controller) trở nên dễ dàng hơn.

Ví dụ: Có thể kiểm tra logic nghiệp vụ trong Model mà không cần giao diện View.

**Tính linh hoạt và mở rộng (Flexibility and Scalability)**

MVC cho phép thêm hoặc sửa đổi các thành phần mà không làm ảnh hưởng đến các phần khác.

Ví dụ: Thêm một View mới để hỗ trợ giao diện người dùng trên thiết bị khác.

**Hỗ trợ phát triển đồng thời (Parallel Development)**

Các nhóm phát triển có thể làm việc song song trên các thành phần khác nhau (Model, View, Controller) mà không gây xung đột.