**1. Tìm hiểu đặc điểm chính của mô hình MVC**

**MVC (Model - View - Controller)** là một kiến trúc phần mềm được sử dụng rộng rãi trong phát triển ứng dụng, đặc biệt là ứng dụng web. Mô hình này chia ứng dụng thành 3 thành phần riêng biệt nhằm **tách biệt logic xử lý, giao diện và điều khiển**, giúp dễ dàng bảo trì, mở rộng và kiểm thử.

### \* ****Model – Mô hình dữ liệu****

* Chứa **dữ liệu ứng dụng** và **logic nghiệp vụ**.
* Là phần duy nhất **trực tiếp thao tác với CSDL** (MySQL, Oracle, SQLite...).
* Thường là các lớp **Java Beans, DTO, Entity**.
* Không liên quan gì đến giao diện người dùng.

**Ví dụ:**  
Lớp NguoiDung chứa các thuộc tính: tenDangNhap, matKhau, vaiTro, kèm các phương thức kiểm tra hợp lệ.

**\*View – Giao diện**

* Là phần **hiển thị dữ liệu** từ Model ra màn hình.
* Không chứa bất kỳ logic xử lý dữ liệu nào.
* Có thể là **JSP**, **HTML**, **Thymeleaf**, **Swing GUI**, v.v.
* View gửi yêu cầu tới Controller.

**Ví dụ:**  
Trang dangnhap.jsp có form nhập tên đăng nhập và mật khẩu.

**Controller – Bộ điều khiển**

* Nhận các hành động từ người dùng (click, submit,...).
* Gửi dữ liệu tới Model xử lý → lấy kết quả → chuyển sang View để hiển thị.
* Là nơi chứa logic điều hướng (navigation/forward/redirect).

**Ví dụ:**  
Lớp DangNhapServlet nhận yêu cầu POST từ form, kiểm tra thông tin đăng nhập và chuyển hướng đến View tương ứng.

\*Các thành phần chính của MVC:

| **Thành phần** | **Mô tả** | **Vai trò** |
| --- | --- | --- |
| **Model** | Lớp đại diện cho dữ liệu, xử lý logic nghiệp vụ, tương tác với cơ sở dữ liệu. | Quản lý dữ liệu và quy tắc nghiệp vụ. |
| **View** | Lớp hiển thị dữ liệu cho người dùng (UI). | Giao diện người dùng, hiển thị thông tin. |
| **Controller** | Lớp xử lý các yêu cầu từ người dùng, cập nhật Model và View tương ứng. | Điều phối giữa View và Model. |

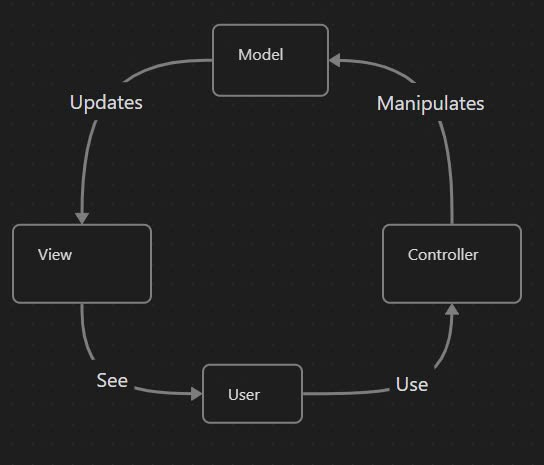
**2. Vẽ kiến trúc MVC là gì?**

Việc "vẽ kiến trúc MVC" nghĩa là biểu diễn **mối quan hệ giữa Model - View - Controller** bằng **sơ đồ kiến trúc**, giúp hiểu rõ luồng xử lý và vai trò của từng phần trong ứng dụng.

Mục đích của việc này là để:

* Thấy được **sự phân chia trách nhiệm rõ ràng** giữa 3 thành phần: Model, View và Controller
* Hiểu rõ **luồng dữ liệu và tương tác giữa các thành phần**
* Dễ dàng **trình bày trong báo cáo hoặc thuyết trình**

**3. SƠ ĐỒ CÂY**



4. ƯU ĐIỂM & NHƯỢC ĐIỂM CỦA MÔ HÌNH MVC

| **Ưu điểm** | **Nhược điểm** |
| --- | --- |
| Tách biệt rõ ràng giữa xử lý và hiển thị | Ban đầu hơi phức tạp đối với người mới |
| Dễ dàng mở rộng và bảo trì từng phần riêng biệt | Có thể gây dư thừa file nếu dự án nhỏ |
| Cho phép phát triển song song (nhiều người cùng làm Model/View/Controller) | Debug lỗi khó hơn nếu không quản lý tốt luồng dữ liệu |
| Tái sử dụng code hiệu quả, dễ viết Unit Test | Cần hiểu rõ quy trình request/response và lifecycle web |

\*TỔNG KẾT

| **Thành phần** | **Vai trò trong ví dụ BMI** |
| --- | --- |
| Model | BmiModel.java – Tính toán BMI |
| View | form.jsp, ketqua.jsp – Giao diện nhập liệu và kết quả |
| Controller | BmiServlet.java – Nhận dữ liệu từ form, gọi Model, chuyển kết quả |