

Quản lý quy trình phần mềm

Software Architecture

Website đọc báo điện tử

GVHD: Ngô Huy Biên



Bộ môn Công nghệ phần mềm

Khoa Công nghệ thông tin

Đại học Khoa học tự nhiên TP

1. Thông tin thành viên nhóm.

MSSV	Họ Tên
22424012	Dương Thị Ánh Nguyệt
22424015	Lý Mai Tiến Tài
22424016	Hoàng Ngọc Thiện
22424022	Trương Bửu Khang
22424023	Hồ Tuấn Kiệt
22424027	Nguyễn Thị Phương Thảo
22424028	Trần Nguyên Thảo
22424036	Nguyễn Thanh Vinh
21424070	Nguyễn Võ Quang Vinh

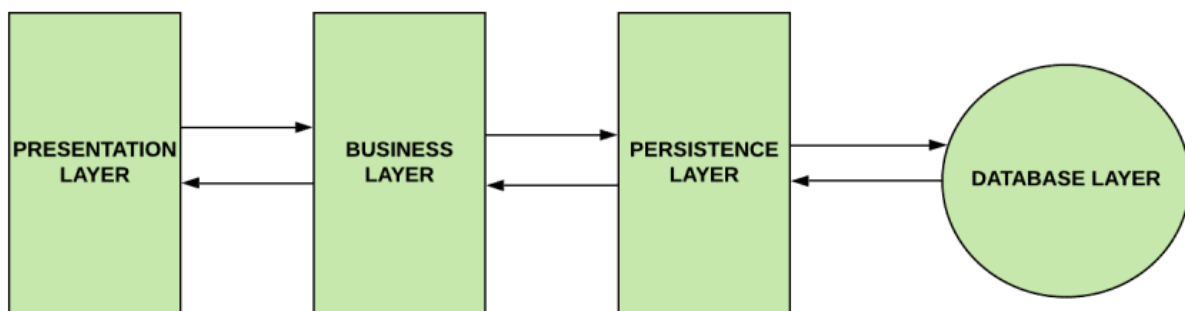
2. Mục tiêu của dự án.

Dự án được thiết kế để cung cấp một trải nghiệm đọc báo điện tử dễ dàng, nhanh chóng và phù hợp với người dùng, đồng thời cho phép quản trị viên quản lý và cập nhật nội dung một cách nhanh chóng, hiệu quả, dễ dàng.

3. Kiến trúc tổng quan.

a) Backend

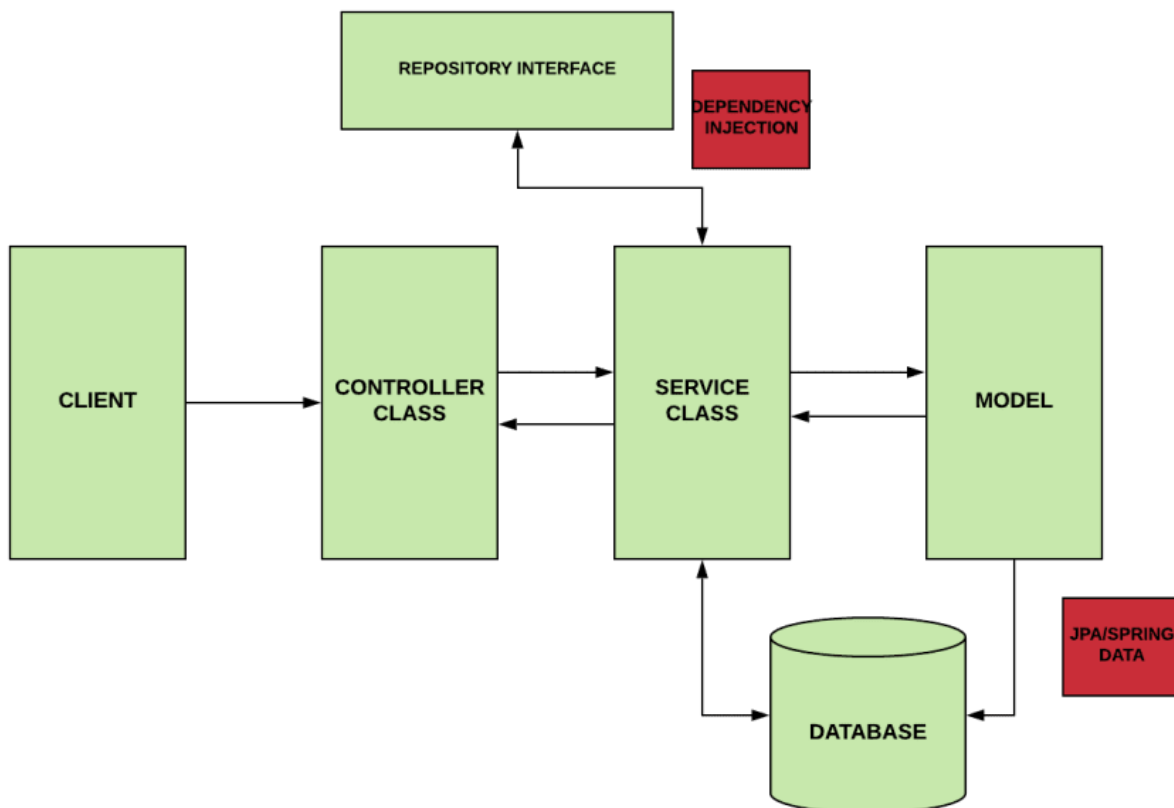
❖ Sử dụng *Spring Boot* của *Spring Framework* với kiến trúc 4 lớp:



- *Presentation:*

- Thực hiện authentication.
 - Chuyển đổi dữ liệu JSON thành đối tượng (và ngược lại).
 - Xử lý các HTTP request.
 - Truyền authentication tới lớp bussiness.
- *Business:*
- Thực hiện validation.
 - Thực hiện authorization (uỷ quyền).
 - Xử lý các logic và quy tắc nghiệp vụ.
- *Persistence:*
- Chứa các logic lưu trữ
 - Lấy các đối tượng và chuyển đổi thành các hàng trong database (và ngược lại).
- *Database:*
- Thực hiện các tác vụ với database (chủ yếu là CRUD).

❖ Workflow của **Spring Boot**:

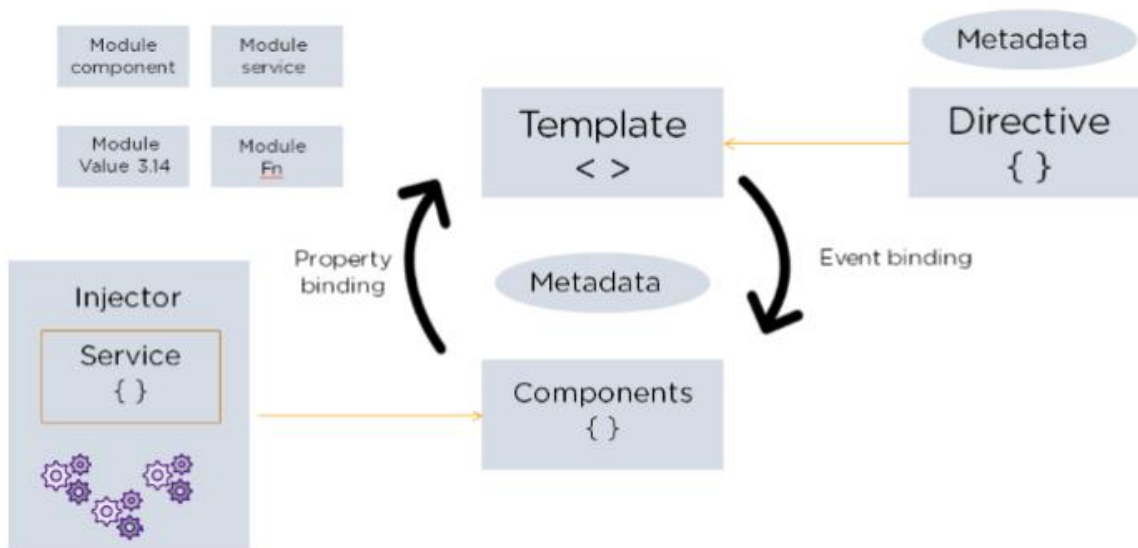


Cách hoạt động như sau:

1. Client tạo một HTTP request.
2. Class Controller nhận HTTP request.
3. Controller hiểu loại request và xử lý nó.
4. Nếu cần, nó gọi class Service.
5. Class Service sẽ xử lý logic nghiệp vụ, với dữ liệu từ database.
6. Nếu mọi thứ hoạt động tốt, ta sẽ trả về một trang JSP(JavaServer Page).

b) Frontend

❖ Sử dụng Framework của Javascript là **Angular**



- **Module:**
 - Chia ứng dụng thành các phần chức năng riêng biệt, giúp quản lý code dễ dàng.
 - Import và export các thành phần như components, directives, và services để chia sẻ chúng giữa các module khác nhau.
- **Component:**
 - Là thành phần cơ bản của Angular, đại diện cho một phần giao diện (UI) và logic ứng dụng.
 - Có thể sử dụng trong template của các component khác.
- **Templates:**
 - Xác định cách giao diện của component sẽ được render lên giao diện (UI).
 - Sử dụng ng-binding để liên kết dữ liệu từ component và hiển thị chúng trên giao diện (UI).
- **Metadata:**
 - Metadata cho Angular biết cách xử lý một lớp.

- Sử dụng để trang trí lớp để nó có thể cấu hình hành vi mong đợi của một lớp.
- *Service*:
 - Lưu trữ các logic và xử lý dữ liệu.
 - Được sử dụng để chia sẻ dữ liệu và logic giữa các component.
- *Dependency Injector*:
 - Cách Angular quản lý và cung cấp các service cho các component.
 - Cho phép giữ cho các lớp component của mình hoạt động rõ ràng và hiệu quả.

4. Quản lý dự án.

- ❖ Sử dụng các phương pháp phát triển phần mềm **Scrum** để quản lý dự án và theo dõi tiến độ.
- ❖ Thực hiện kiểm tra và kiểm thử liên tục để đảm bảo chất lượng mã nguồn dự án và tích hợp việc sửa chữa và cập nhật một cách nhanh chóng, dễ dàng.