**BẢNG ĐÁNH GIÁ ĐIỂM THUYẾT TRÌNH**

**MÔN LẬP TRÌNH JAVA NÂNG CAO**

**LỚP CTK44-PM**

|  |  |
| --- | --- |
| **Thông tin đề tài thuyết trình** | **Thông tin sinh viên thực hiện đánh giá** |
| Tên đề tài: Cách tổ chức và chia sẻ dữ liệu của Spring Boot | MSSV của sinh viên đánh giá: 2012353 |
| Điểm tổng kết: Ngày trình bày: 25/04/2023 | Họ tên sinh viên & ký xác nhận: Đoàn Cao Nhật Hạ |
| ***Thông tin sinh viên thuyết trình*** |
| MSSV: 2014469 - 2012264  Họ tên: Trần Ngọc Hoàng Long - Trần Nhật Duật |

1. **ĐÁNH GIÁ THUYẾT TRÌNH**

**Thang điểm: Kém 0-5; Trung bình: 5-7; Khá 7-8; Tốt 8-9; Xuất sắc: 9-10**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Bảng chấm điểm nhóm thuyết trình** | | | | | | |
| Các chủ đề chính có được xác định? (A) | Slide thuyết trình có thiết kế tốt (B) | Giao tiếp với người nghe bằng ánh mắt (C) | Giao tiếp với người nghe bằng giọng nói (D) | Khả năng trình bày nội dung (E) | Thảo luận & Hỏi đáp (F) | Điểm trung bình = (A+B+C+D+E+F)/6 |
| **9** | **9** | **9** | **9** | **8,8** | **8,8** | **9** |

1. **BÀI TẬP NẮM KIẾN THỨC**

Dựa vào nội dung vừa trình bày, trả lời các câu hỏi sau:

1. Trình bày tóm tắt nội dung anh/ chị đã nắm được về chủ đề vừa trình bày?

* Spring là một framework mã nguồn mở, cung cấp cấu trúc để phát triển các ứng dụng java có hiệu năng cao
* Spring tổ chức theo module nên dễ kiểm thử và tái sử dụng lại code
* Tính năng cốt lõi:

+ Phát triển Java Desktop, ứng dụng mobile và web.

+ Mục tiêu chính của Spring là giúp phát triển các ứng dụng J2EE một cách dễ dàng hơn dựa trên mô hình sử dụng POJO (Plain Old Java Object)

* "Java EE" tên đầy đủ là "Java Enterprise Edition" dịch sang tiếng việt có nghĩa là là "Java Phiên Bản Doanh Nghiệp", nó là một nền tảng (platform) dành cho việc xây dựng các ứng dụng (application) cấp doanh nghiệp.
* Java beans là các lớp java thuần dùng để thể hiện các thực thể trong các chương trình java
* Một lớp java been sẽ có 3 đặc điểm như sau về mặt cú pháp

+ Là một public class

+ Có các thuộc tính private

+ Ứng với mỗi thuộc tính lớp sẽ cung cấp cặp phương thức setter và getter để truy cập và thao tác với giá trị của từng thuộc tính

+ Setter: là các phương thức cho phép thiết lập giá trị của thuộc tính

+ Getter là các phương thức cho phép truy cập và lấy về giá trị của thuộc tính

* Spring MVC là một framework dựa trên Spring, được thiết kế dành cho xây dựng ứng dụng nền tảng web.
* Spring MVC dựa trên mô hình MVC (Model-View-Controller)
* Trung tâm chính của Spring MVC là Dispatcher Servlet là một front controller. Servlet điều phối gọi một đối tượng để xử lý yêu cầu duy nhất: ánh xạ trình xử lý một yêu cầu duy nhất: ánh xạ trình xử lý handler mapping, controller, view resolver và view: cách nó hoạt động theo thứ tự là:

1: Ánh xạ handler mapping xác định URL nào được xử lý bởi phương thức xử lý nào trong class controller nào

2: một controller nhận yêu cầu, đọc đầu vào, gọi đối tượng nghiệp vụ xử lý đầu vào và chọn tên dạng view logic cho yêu cầu

3: trình phân giải View Resolver xác định tên của view nào thì ánh xạ tới view template (HTML, JSP, …)

4: Controller chọn xem template nào thích hợp thì chuyển dữ liệu model và tạo phản hồi

* Ưu điểm:

+ Các tầng trong Spring MVC độc lập nên việc unit test trở nên dễ dàng hơn

+ Phần view có thể tích hợp nhiều Framework về UI như JSF, Thymeleaf

+ Hỗ trợ cả Annotation và  XML config giúp việc phát triển code nhanh hơn và clean hơn

+ Cung cấp việc phân chia một cách rõ ràng, linh hoạt giữa controller, service, data access layer

* Nhược điểm:  Khi sử dụng Spring MVC, chúng ta cần cấu hình
  + Component scan
  + Dispatcher Servlet
  + View resolver
  + Web jars

Định nghĩa: Spring Boot là một dự án phát triển bởi JAV (ngôn ngữ java) trong hệ sinh thái Spring framework. Nó giúp cho các lập trình viên chúng ta đơn giản hóa quá trình lập trình một ứng dụng với Spring, chỉ tập trung vào việc phát triển business cho ứng dụng.

5 bước thiết kế trang web dùng spring boot:

* Tạo một project sử dụng Maven với các dependency cần thiết của Spring MVC và Servlet API.
* Một tập tin web.xml để khai báo DispatcherServlet của Spring MVC.
* Một tập tin cấu hình của Spring MVC.
* Một class Controller trả về một trang “Hello World” khi có request đến.
* Cuối cùng là phải có một web server dùng để triển khai ứng dụng lên chạy.

Một số tính năng nổi bật của Spring Boot:

* Tạo các ứng dụng Spring độc lập
* Nhúng trực tiếp Tomcat, Jetty hoặc Undertow (không cần phải deploy ra file WAR)
* Các starter dependency giúp việc cấu hình Maven đơn giản hơn
* Tự động cấu hình Spring khi cần thiết
* Không sinh code cấu hình và không yêu cầu phải cấu hình bằng XML …
* Các đặc tính cơ bản của Spring Boot bao gồm những khía cạnh cụ thể sau đây:
* SpringApplication: Đây là một Class giúp để khởi chạy những ứng dụng bắt đầu từ hàm main. Để bắt đầu sử dụng, bạn chỉ cần thiết lập các method run.
* Profiles: Spring Boot Profiles cung cấp phương pháp để phân chia các cấu hình riêng biệt cho từng môi trường. Các annotation bao gồm @Configuration và@Component sẽ được đánh dấu profiles nhằm mục đích giới hạn môi trường cũng như thời điểm tải lên.
* Externalized Configuration: Tính năng này cho phép người dùng có cấu hình từ bên ngoài. Vậy nên một ứng dụng có thể dễ dàng tạo dựng và vận hành trên nhiều môi trường khác nhau. Để thực hiện Externalized Configuration bạn có thể gọi các YAML, file properties cùng các tham số command line hoặc là tham biến môi trường.
* Logging: Tất cả những chức năng log nội bộ đều được spring boot dùng thông qua Logging và được quản lý một cách mặc định. Vậy nên không cần thiết phải sửa các dependency logging hoặc các tùy biến customization không được yêu cầu.

Mô hình:

* 1. Mô hình ba lớp (three tier):
* Đây là mô hình tổ chức source code rất phổ biến trong Spring Boot. Cụ thể, ứng dụng được chia làm 3 tầng (tier hoặc layer) như sau:
* Presentation layer: tầng này tương tác với người dùng, bằng View, Controller (trong MVC) hoặc API (nếu có).
* Business logic layer: Chứa toàn bộ logic của chương trình, các đa số code nằm ở đây
* Data access layer: Tương tác với database, trả về kết quả cho tầng business logic
* Trong Spring Boot, thì có một số thành phần đại diện cho từng lớp:
* Service: chứa các business logic code
* Repository: đại diện cho tầng data access
  1. Mô hình MVC

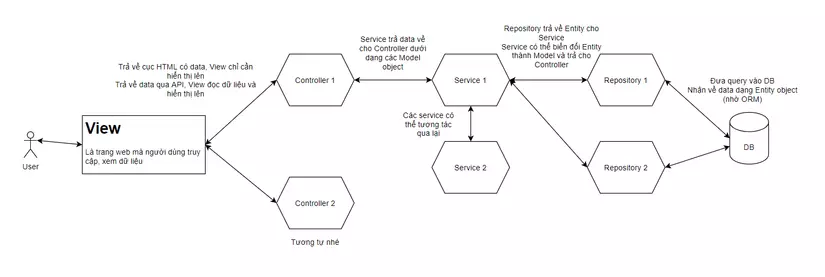
Do Spring Boot chỉ là wrapper cho Spring, chúng ta vẫn sử dụng ngầm các module Spring khác bên dưới, ví dụ như Spring MVC. Và tất nhiên khi dùng Spring MVC thì sẽ tuân theo mô hình MVC rồi .

Mô hình MVC:

Cụ thể, nó chia tầng presentation làm 3 phần:

* Model: các cấu trúc dữ liệu của toàn chương trình, có thể đại diện cho trạng thái của ứng dụng
* View: lớp giao diện, dùng để hiển thị dữ liệu ra cho user xem và tương tác
* Controller: kết nối giữa Model và View, điều khiển dòng dữ liệu
* Dữ liệu từ Model qua Controller sau đó được gửi cho View hiển thị ra. Và ngược lại, khi có yêu cầu mới từ View, thì sẽ qua Controller thực hiện thay đổi dữ liệu của Model.

1.3 Sơ đồ luồng đi:



Đầu tiên, user sẽ vào View để xem, tương tác

Khi user bắt đầu load dữ liệu (ví dụ click nút Reload), thì 1 request từ View gửi cho Controller

Controller nhận được yêu cầu, bắt đầu đi hỏi ông Service

Service nhận được yêu cầu từ Controller, đối với các code đơn giản có thể tính toán và trả về luôn. Nhưng các thao tác cần đụng tới database thì Service phải gọi Repository để lấy dữ liệu trong DB

Repository nhận được yêu cầu từ Service, sẽ thao tác với DB. Data lấy ra trong DB được hệ thống ORM (như JPA hoặc Hibernate) mapping thành các object (trong Java). Các object này gọi là Entity.

Và bây giờ sẽ là đi ngược lại trả về cho user:

Service nhận các Entity được Repository trả về, biến đổi nó. Biến đổi ở đây là có thể thực hiện tính toán, thêm bớt các field,... và cuối cùng biến Entity thành Model. Model sẽ được trả lại cho Controller.

Controller nhận được Model, nó sẽ return cho View. Có 2 cách, một là dùng template engine pass dữ liệu Model vào trang HTML, rồi trả về cục HTML (đã có data) cho client. Cách 2 là gửi qua API, View tự parse ra và hiển thị tương ứng (hiển thị thế nào tùy View).

Khi View hiển thị xong, user sẽ thấy danh sách user hiện lên trang web.

1. Anh/ chị hãy đưa ra các nhận xét và góp ý cho sinh viên vừa trình bày?