**Microservices with Spring Boot 3 and Spring Cloud**

**Creating Microservices for account and loan**

**//HateoasBookController.java**

package controller;

import model.Book;

import repository.BookRepository;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.hateoas.Link;

import org.springframework.web.bind.annotation.\*;

import java.util.List;

import java.util.stream.Collectors;

import static org.springframework.hateoas.server.mvc.WebMvcLinkBuilder.\*;

@RestController

@RequestMapping("/api/hateoas-books")

public class HateoasBookController {

@Autowired

private BookRepository repo;

@GetMapping

public List<Book> getAll() {

return repo.findAll().stream().map(book -> {

book.add(linkTo(methodOn(HateoasBookController.class).getBook(book.getId())).withSelfRel());

book.add(linkTo(methodOn(HateoasBookController.class).getAll()).withRel("all-books"));

return book;

}).collect(Collectors.toList());

}

@GetMapping("/{id}")

public Book getBook(@PathVariable Long id) {

Book book = repo.findById(id).orElseThrow();

book.add(linkTo(methodOn(HateoasBookController.class).getBook(id)).withSelfRel());

book.add(linkTo(methodOn(HateoasBookController.class).getAll()).withRel("all-books"));

return book;

}

@PostMapping

public Book create(@RequestBody Book book) {

Book saved = repo.save(book);

saved.add(linkTo(methodOn(HateoasBookController.class).getBook(saved.getId())).withSelfRel());

return saved;

}

}

**//Book.java**

package model;

import jakarta.persistence.\*;

import org.springframework.hateoas.RepresentationModel;

@Entity

public class Book extends RepresentationModel<Book> {

@Id

@GeneratedValue

private Long id;

private String title;

private String author;

public Book() {}

public Book(String title, String author) {

this.title = title;

this.author = author;

}

// Getters & setters

public Long getId() { return id; }

public void setId(Long id) { this.id = id; }

public String getTitle() { return title; }

public void setTitle(String title) { this.title = title; }

public String getAuthor() { return author; }

public void setAuthor(String author) { this.author = author; }

}

**//BookRepository.java**

package repository;

import model.Book;

import org.springframework.data.jpa.repository.JpaRepository;

public interface BookRepository extends JpaRepository<Book, Long> { }

**//HateoasApp.java**

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication(scanBasePackages = {"model", "repository", "controller"})

public class HateoasApp {

public static void main(String[] args) {

SpringApplication.run(HateoasApp.class, args);

}

}

**//BookDtoController.java**

package controller;

import dto.BookRequest;

import dto.BookResponse;

import model.Book;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.web.bind.annotation.\*;

import repository.BookRepository;

import java.util.List;

import java.util.stream.Collectors;

@RestController

@RequestMapping("/api/dto-books")

public class BookDtoController {

@Autowired

private BookRepository repo;

@PostMapping

public BookResponse addBook(@RequestBody BookRequest req) {

Book book = new Book(req.getTitle(), req.getAuthor());

book = repo.save(book);

return new BookResponse(book.getId(), book.getTitle(), book.getAuthor(), "Book created successfully");

}

@GetMapping

public List<BookResponse> getBooks() {

return repo.findAll().stream()

.map(book -> new BookResponse(book.getId(), book.getTitle(), book.getAuthor(), "Success"))

.collect(Collectors.toList());

}

}

**//BookRequest.java**

package dto;

public class BookRequest {

private String title;

private String author;

// Getters & Setters

public String getTitle() { return title; }

public void setTitle(String title) { this.title = title; }

public String getAuthor() { return author; }

public void setAuthor(String author) { this.author = author; }

}

**//BookResponse.java**

package dto;

public class BookResponse {

private Long id;

private String title;

private String author;

private String message;

public BookResponse(Long id, String title, String author, String message) {

this.id = id;

this.title = title;

this.author = author;

this.message = message;

}

// Getters only

public Long getId() { return id; }

public String getTitle() { return title; }

public String getAuthor() { return author; }

public String getMessage() { return message; }

}

**//Book.java**

package model;

import jakarta.persistence.\*;

@Entity

public class Book {

@Id

@GeneratedValue

private Long id;

private String title;

private String author;

public Book() {}

public Book(String title, String author) {

this.title = title;

this.author = author;

}

// Getters & Setters

public Long getId() { return id; }

public void setId(Long id) { this.id = id; }

public String getTitle() { return title; }

public void setTitle(String title) { this.title = title; }

public String getAuthor() { return author; }

public void setAuthor(String author) { this.author = author; }

}

**//BookRepository.java**

package repository;

import model.Book;

import org.springframework.data.jpa.repository.JpaRepository;

public interface BookRepository extends JpaRepository<Book, Long> { }

**//BookDtoApp.java**

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication(scanBasePackages = {"controller", "model", "repository", "dto"})

public class BookDtoApp {

public static void main(String[] args) {

SpringApplication.run(BookDtoApp.class, args);

}

}

**//BookController.java**

package controller;

import model.Book;

import repository.BookRepository;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.web.bind.annotation.\*;

import java.util.List;

@RestController

@RequestMapping("/api/books")

public class BookController {

@Autowired

private BookRepository repo;

@GetMapping

public List<Book> getAll() {

return repo.findAll();

}

@PostMapping

public Book create(@RequestBody Book book) {

return repo.save(book);

}

@PutMapping("/{id}")

public Book update(@PathVariable Long id, @RequestBody Book book) {

Book existing = repo.findById(id).orElseThrow();

existing.setTitle(book.getTitle());

existing.setAuthor(book.getAuthor());

return repo.save(existing);

}

@DeleteMapping("/{id}")

public void delete(@PathVariable Long id) {

repo.deleteById(id);

}

}

**//Book.java**

package model;

import jakarta.persistence.\*;

@Entity

public class Book {

@Id

@GeneratedValue

private Long id;

private String title;

private String author;

// Getters and setters

}

**EXAMPLE RESPONSE:**

{

"id": 1,

"title": "Spring in Action",

"author": "Craig Walls",

"\_links": {

"self": {

"href": "http://localhost:8080/api/hateoas-books/1"

},

"all-books": {

"href": "http://localhost:8080/api/hateoas-books"

}

}

}