**Scenario 15: Online Bookstore - API Documentation with Swagger**

1. **Add Swagger/Springdoc Dependency**

<dependency>

<groupId>org.springdoc</groupId>

<artifactId>springdoc-openapi-ui</artifactId>

<version>1.7.0</version> <!-- Ensure the version is up-to-date -->

</dependency>

1. **Document Endpoints with Annotations**

import io.swagger.v3.oas.annotations.Operation;

import io.swagger.v3.oas.annotations.responses.ApiResponse;

import io.swagger.v3.oas.annotations.responses.ApiResponses;

import io.swagger.v3.oas.annotations.parameters.Parameter;

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

@RestController

@RequestMapping("/api/books")

public class BookController {

@Operation(summary = "Get all books", description = "Returns a list of all books available in the bookstore")

@GetMapping

public List<Book> getAllBooks() {

// ... implementation

}

@Operation(summary = "Get book by ID", description = "Returns a book based on the provided ID")

@ApiResponses(value = {

@ApiResponse(responseCode = "200", description = "Book found"),

@ApiResponse(responseCode = "404", description = "Book not found")

})

@GetMapping("/{id}")

public Book getBookById(

@Parameter(description = "ID of the book to be retrieved", required = true)

@PathVariable Long id) {

// ... implementation

}

@Operation(summary = "Add a new book")

@PostMapping

public Book createBook(@RequestBody Book book) {

// ... implementation

}

@Operation(summary = "Delete a book")

@DeleteMapping("/{id}")

public void deleteBook(@PathVariable Long id) {

// ... implementation

}

}

**3. Generate and Review API Documentation Using Swagger UI**

Once the dependencies are set and the annotations are added, Springdoc will automatically generate Swagger documentation. You can view the API documentation in a user-friendly UI.

**Run your application and visit the Swagger UI page:**

* Swagger UI is typically available at: http://localhost:8080/swagger-ui.html
* Springdoc UI is available at: http://localhost:8080/swagger-ui/index.html

The UI will list all your API endpoints, including their descriptions, parameters, request/response examples, and status codes.

**4. Additional Customization**

import io.swagger.v3.oas.models.OpenAPI;

import io.swagger.v3.oas.models.info.Info;

import org.springframework.context.annotation.Bean;

import org.springframework.context.annotation.Configuration;

@Configuration

public class SwaggerConfig {

@Bean

public OpenAPI customOpenAPI() {

return new OpenAPI()

.info(new Info().title("Bookstore API")

.version("1.0")

.description("Documentation for the Bookstore API"));

}

}