Swagger com SpringDoc

Certified Developer
The Ultimate Tech Degree

DigitalHouse >

Exemplo de controller

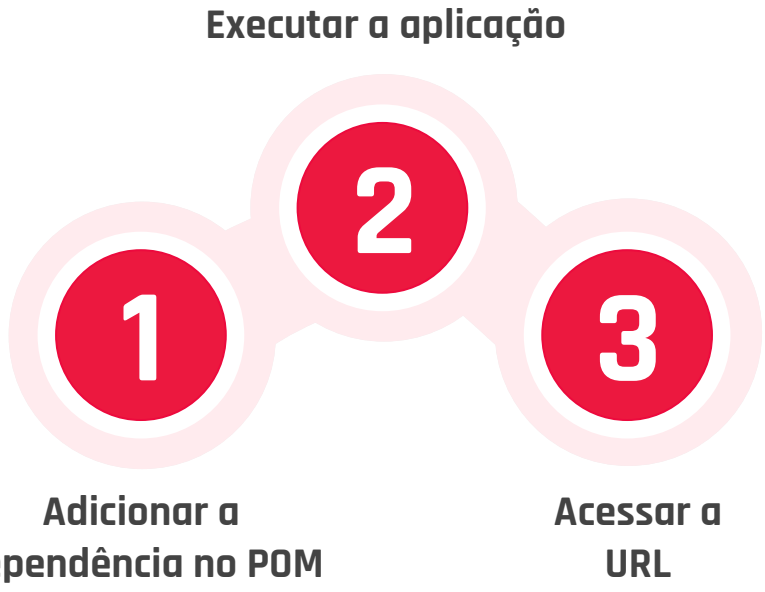
Suponha que temos um controller em nossa aplicação para gerenciar livros.

```
@RestController
@RequestMapping("/api/book")
public class BookController {

    @Autowired
    private BookRepository repository;

    @GetMapping("/{id}")
    public Book findById(@PathVariable long id) {
        return repository.findById(id)
            .orElseThrow(() -> new BookNotFoundException());
    }
```

```
    @GetMapping("/")
    public Collection<Book> findBooks() {
        return repository.getBooks();
    }

    @PutMapping("/{id}")
    @ResponseStatus(HttpStatus.OK)
    public Book updateBook(
      @PathVariable("id") final String id, @RequestBody final Book book) {
        return book;
    }
}
```

Passos

Agora veremos um exemplo passo a passo de como usar a biblioteca do SpringDoc.



1 - Adicionar a dependência no POM

```
<dependency>
    <groupId>org.springdoc</groupId>
    <artifactId>springdoc-openapi-ui</artifactId>
    <version>1.5.2</version>
</dependency>
```

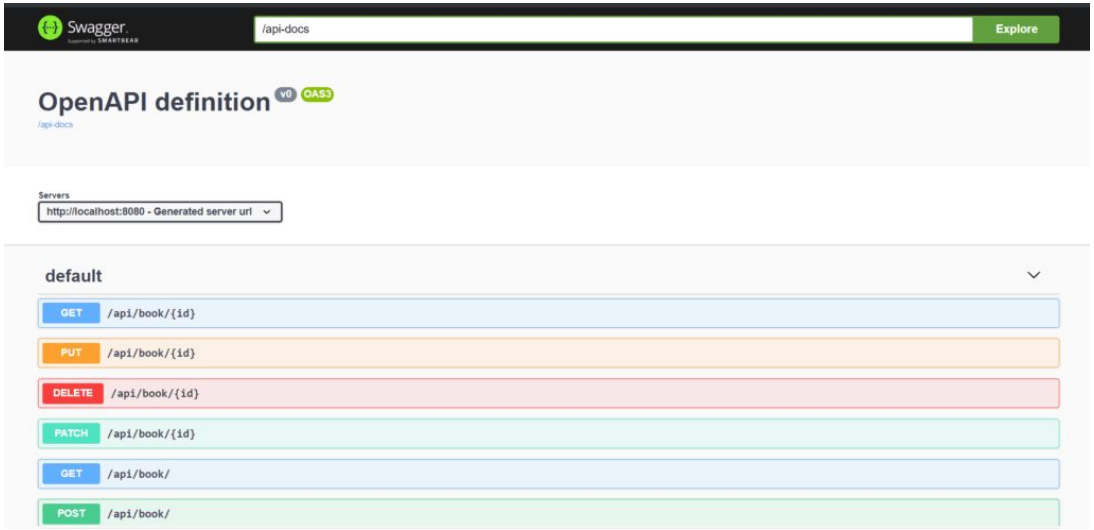
2 - Executar a aplicação

Executamos nossa aplicação e nos dirigimos a URL de Swagger.

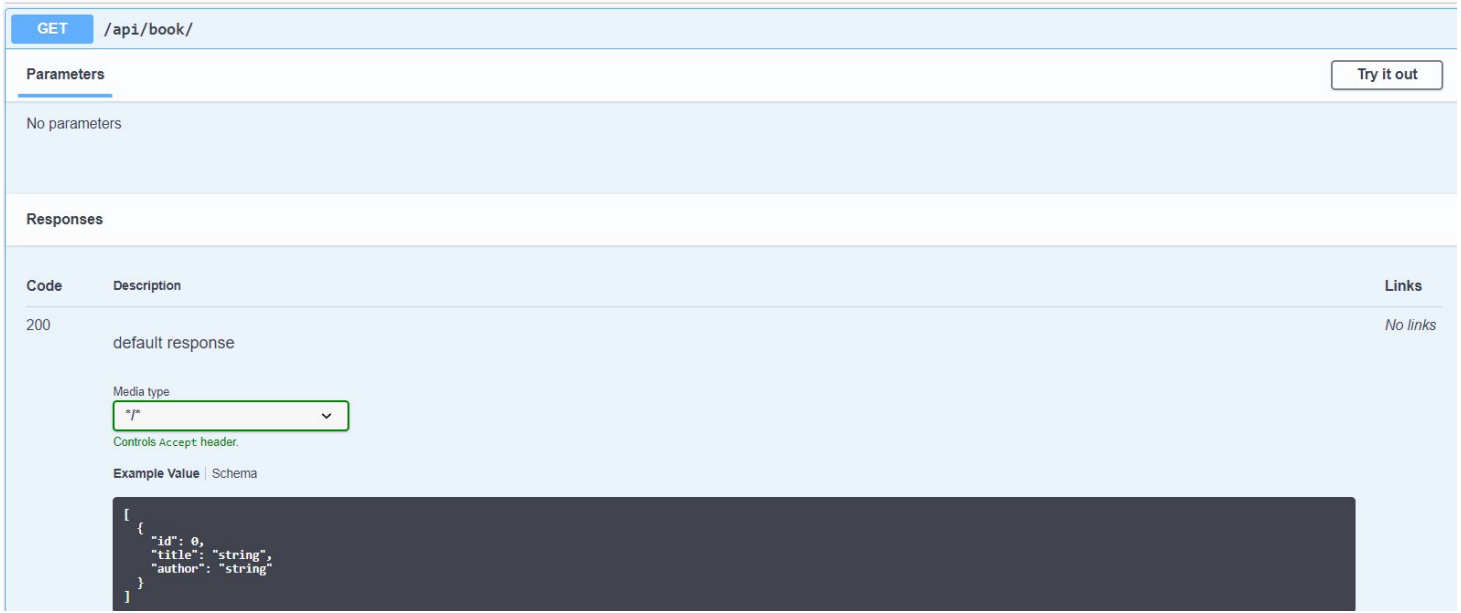
Por padrão, a interface criada pelo Swagger estará em:
http://localhost:8080/swagger-ui.html

3 - Acessar a URL

Na URL, você encontrará a documentação da nossa API em /api-docs.



Nesta tela podemos ver todos os endpoints de nossa aplicação e os detalhes de como usar cada um.



Requisitos nos campos

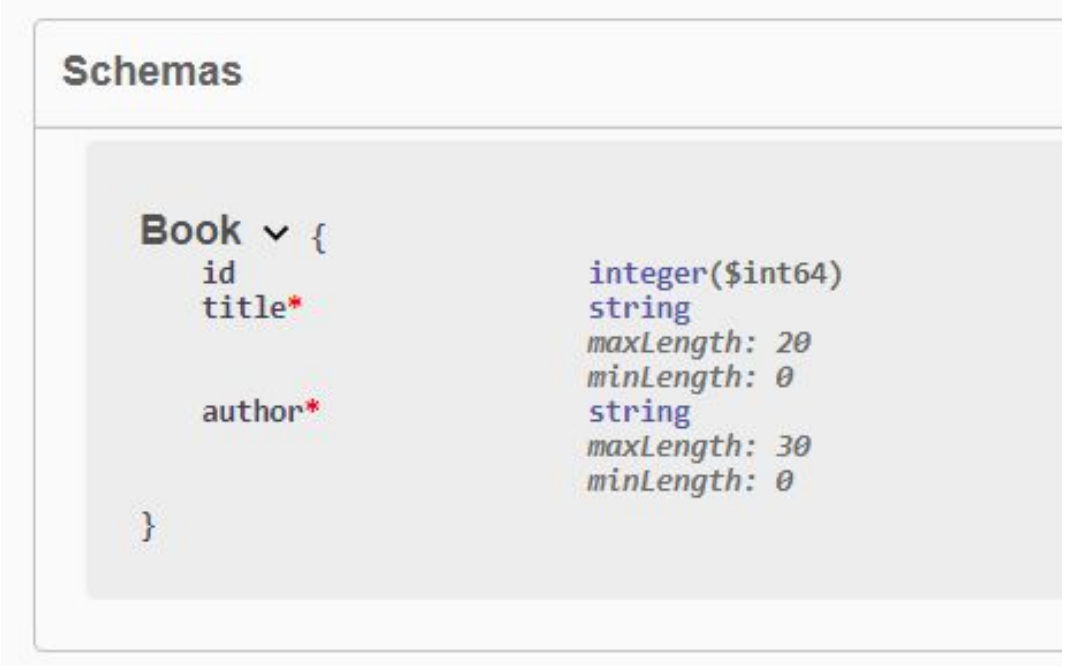
Podemos adicionar os requisitos na documentação do Swagger utilizando anotações em nossas entidades, com @NotNull, @NotBlank, @Size, @Min e @Max.

- **@NotNull** indica que o campo não pode ser nulo;
- **@NotBlank** indica que o campo deve ter 1 ou mais caractere;
- **@Size** é usado para indicar o tamanho máximo de caractere que o campo recebe.

Por exemplo:

```
public class Book {

    private long id;

    @NotBlank
    @Size(min = 0, max = 20)
    private String title;

    @NotBlank
    @Size(min = 0, max = 30)
    private String author;
}
```

No Swagger veríamos:



DigitalHouse>