

FIAP GRADUAÇÃO

SISTEMAS DE INFORMAÇÃO

MICROSERVICE AND WEB ENGINEERING

Prof^a. Aparecida Castello Rosa
profaparecida.rosa@fiap.com.br

I Agenda

- Continuar com o projeto produto-mvc
- CRUD – *Update* – Atualizar produto
- CRUD – *Delete* – Deletar produto
- View para editar produto

Objetivos

- Continuar com o projeto produto-mvc
- Criar funcionalidade de editar produto
- Criar View para editar produto
- Criar funcionalidade para deletar produto

← ↻ 🏠 ⓘ localhost:8080/produtos/listar 🔍 🔊 ☆ 📄 ⚙️ 🌐

📁 Software antivírus e... 📁 Conta da Microsoft... 📁 UNIVERSIDADE 📁 API 📁 Cursos 📁 Hospital 📄 manual [Visualg3] 📁 Melhores Cursos O... 🗣️ Google Tradutor > 📁

FIAP Home Cadastrar Produtos

Lista de Produtos

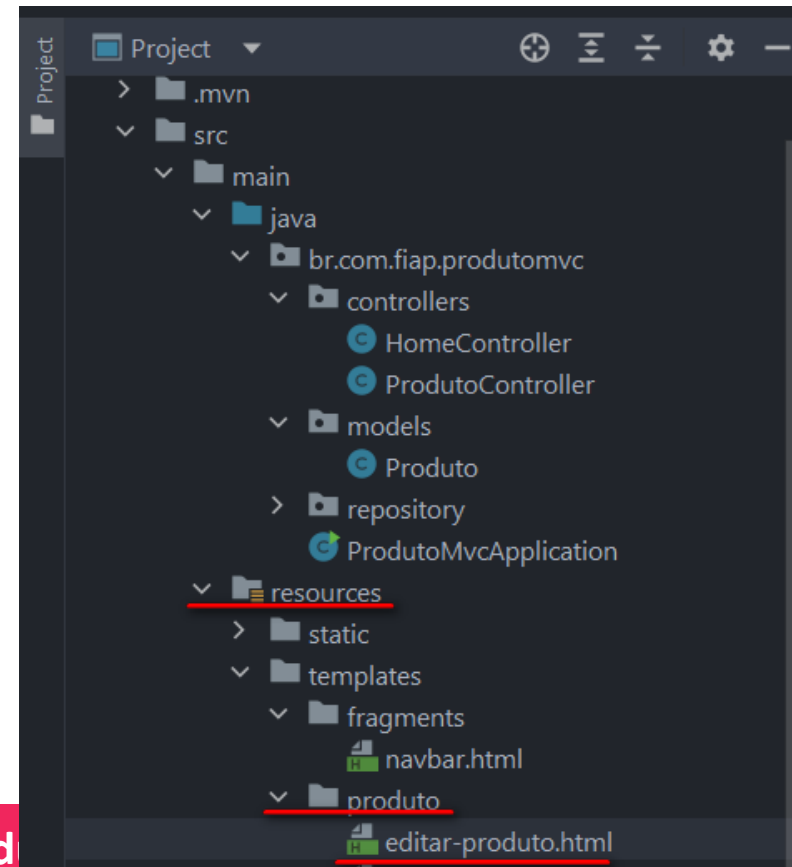
Nome	Descrição	Valor	Ações	
Mouse Microsoft	Mouse sem fio	250.0	Excluir	Editar
Smartphone Samsung Galaxy A54 5G	Samsung Galaxy A54 5G	1799.0	Excluir	Editar
Smart TV	Smart TV LG LED 65 polegadas	3999.0	Excluir	Editar

Incluir th:href tag <a>Editar - listar-produtos.html

```
listar-produtos.html x
34 <td>
35     <a class="btn btn-danger btn-xs"> Excluir</a>
36 </td>
37 <td>
38     <a class="btn btn-primary btn-xs"
39     th:href="@{/produtos/editar/{id}(id=${produto.id})}"
40     > Editar</a>
41 </td>
```

■ Criar View editar-produto

- Baixar o arquivo editar-produto.html anexado a aula para complementar o código. Essa View é muito parecida com a View novo-produto.html.



Alterar View editar-produto

```
Project
editar-produto.html x
22 <!-- th:object faz referencia ao obj - th:object="${produto}" - é o objeto enviado pelo controller - chave
23 <form action="#" th:action="@{/produtos/editar/{id}(id=${produto.id})}" th:object="${produto}" method="post">
24 <div class="row mb-3">
```


Parâmetros nas classes Controllers

Anotação MVC (request)	Objetivo
@PathVariable	Extrair da URL um parâmetro incluído como path da URL
@RequestParam	Capturar um parâmetro de consulta (query param) GET

@PathVariable - usado quando o valor da variável é passado diretamente na URI, exemplo:

```
http://localhost:8080/produtos/editar/1
```

@RequestParam - usado para capturar vários parâmetros que são passado após o sinal de interrogação (?) na URL e que não são partes da URI em si, exemplo:

```
http://localhost:8080/produtos?size=3&page=0?sort=nome
```

URI: Uniform Resource Identifier

URL: Uniform Resource Locator



Basicamente uma URI identifica e uma URL localiza um recurso, entretanto, localizadores também são identificadores, ou seja, as URLs são um subconjunto do conjunto de URIs.

Exercício

Entendendo o Optional

```
@GetMapping("/editar/{id}")
public String editarProduto(@PathVariable("id") Long id, Model model ){
    // faz um select no DB pelo id do produto
    // passado pela URL
    // se não existir, lança exception
    // expressão regular
    Produto produto = repository.findById(id);
}
```

Required type: Produto

Provided: Optional <br.com.fiap.prodotomvc.models.Produto>

Change variable 'produto' type to 'Optional<Produto>' Alt+Shift+Enter More actions... Alt+Enter

```
org.springframework.data.repository.CrudRepository<T, ID>
public abstract java.util.Optional<T> findById(
    ID id
)
```

Retrieves an entity by its id.

Params: **id** – must not be null.

Returns: the entity with the given id or Optional#empty() if none found.

Throws: IllegalArgumentException – if id is null.

Maven: org.springframework.data:spring-data-commons:3.2.2 (spring-data-commons-3.2.2.jar)

Adicionar o método GET editarProduto no ProdutoController

```
ProdutoController.java x
62 //URL - localhost:8080/produtos/editar/1
63 @GetMapping("/editar/{id}")
64 // @PathVariable - extrai da URL um parâmetro incluído com parte da URL
65 @ public String editarProduto(@PathVariable("id") Long id, Model model ){
66     // faz um select no DB pelo id do produto passado pela URL
67     // se não existir, lança exception
68     // expressão regular
69     Produto produto = repository.findById(id).orElseThrow(
70         () -> new IllegalArgumentException("Produto inválido - id: " + id)
71     );
72     model.addAttribute("produto", produto); //devolve o objeto para a View
73     return "/produto/editar-produto"; // View
74 }
```

■ Outra forma de tratar Optional

```
@GetMapping("/editar/{id}")
@Transactional(readonly = true)
public String editarProduto(@PathVariable ("id") Long id, Model model ){

    Optional<Produto> produto = repository.findById(id);
    if(produto.isEmpty()){ // ou if(! produto.isPresent())
        throw new IllegalArgumentException("Produto inválido - id: " + id);
    }

    model.addAttribute("produto", produto.get());
    return "/produto/editar-produto";
}
```

Testar aplicação

localhost:8080/produtos/listar

FIAP Home Cadastrar Produtos

Lista de Produtos

Nome	Descrição	Valor	Ações	
Mouse Microsoft	Mouse sem fio	250.0	Excluir	Editar
Smartphone Samsung Galaxy A54 5G	Samsung Galaxy A54 5G	1799.0	Excluir	Editar
Smart TV	Smart TV LG LED 65 polegadas	3999.0	Excluir	Editar

Alteração de Produto

Nome: Smart TV

Descrição: Smart TV LG LED 65 polega

Valor: 3999.0

Alterar

Testar aplicação

FIAP Home Cadastrar Produtos

Alteração de Produto

Nome: Smart TV - LED

Descrição: Smart TV LG LED 85 polega

Valor: 5500

Alterar

localhost:8080/produtos/editar/3

Software antivírus e... Conta da Microsoft... UNIVERSIDADE API Cursos Hospital

Whitelabel Error Page

This application has no explicit mapping for /error, so you are seeing this as a fallback.

Fri Feb 23 18:24:27 BRT 2024

There was an unexpected error (type=Method Not Allowed, status=405).

Method 'POST' is not supported.

Exercício

Adicionar o método **POST** `editarProduto` no `ProdutoController`

import org.springframework.transaction.annotation.Transactional;
Transacional deve ser do Spring

```
ProdutoController.java x
77
80 //URL - localhost:8080/produtos/editar/id
81 @PostMapping("/editar/{id}")
82 @Transactional
83 @ public String editarProduto(@PathVariable("id") Long id,
84                               @Valid Produto produto,
85                               BindingResult result){
86     if(result.hasErrors()){
87         produto.setId(id);
88         return "/produto/editar-produto";
89     }
90     repository.save(produto); //salva no DB
91     return "redirect:/produtos/listar"; //chama a View
92 }
```

Testar aplicação

FIAP Home Cadastrar Produtos

Alteração de Produto

Nome:

Descrição:

Valor:

FIAP Home Cadastrar Produtos

Lista de Produtos

Nome	Descrição	Valor	Ações	
Mouse Microsoft	Mouse sem fio	250.0	<input type="button" value="Excluir"/>	<input type="button" value="Editar"/>
Smartphone Samsung Galaxy A54 5G	Samsung Galaxy A54 5G	1799.0	<input type="button" value="Excluir"/>	<input type="button" value="Editar"/>
Smart TV - LED	Smart TV LG LED 85 polegadas	5500.0	<input type="button" value="Excluir"/>	<input type="button" value="Editar"/>

Atualizar os métodos com a anotação @Transactional

Incluir @Transactional do Spring - class ProdutoController

org.springframework.transaction.annotation.Transactional;

```
@PostMapping("/salvar")
```

```
@Transactional
```

```
public String insertProduto(@Valid Produto produto,  
                             BindingResult result,  
                             RedirectAttributes attributes){
```

```
@GetMapping("/listar")
```

```
@Transactional(readonly = true)
```

```
public String listarProdutos(Model model){
```

```
@GetMapping("/editar/{id}")
```

```
@Transactional(readonly = true)
```

```
// @PathVariable - Extrair da URL um parâmetro incluído como path da URL
```

```
public String editarProduto(@PathVariable Long id, Model model){
```

Delete

Exercício

Incluir th:href tag <a>Excluir - listar-produtos.html

```
listar-produtos.html x
27 <td><span th:text="${produto.descricao}"> Descrição </span></td>
30 <td><span th:text="${produto.valor}"> Valor </span></td>
31 <td>
32     <a class="btn btn-danger btn-xs"
33     th:href="@{/produtos/deletar/{id}(id=${produto.id})}"
34     th:data-confirm-delete="|Deseja excluir ${produto.nome}?|"
35     onclick="if (!confirm(this.getAttribute('data-confirm-delete')))) return false">
36     Excluir</a>
37 </td>
38 <td>
39     <a class="btn btn-primary btn-xs"
40     th:href="@{/produtos/editar/{id}(id=${produto.id})}"> Editar</a>
41 </td>
```

Adicionar o método `deletarProduto` no `ProdutoController`

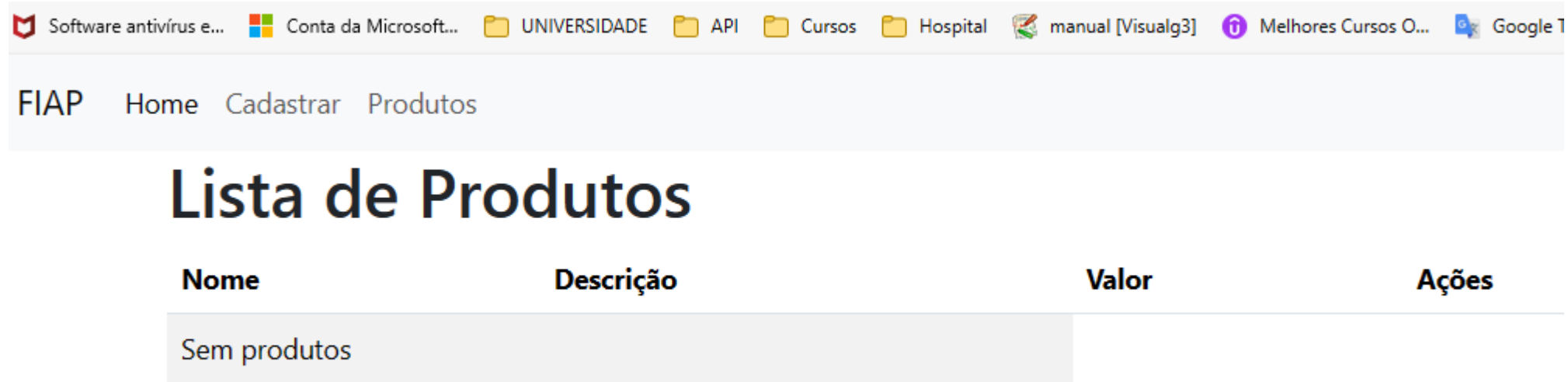
```
ProdutoController.java x
94 //URL - localhost:8080/produtos/deletar/1
95 @GetMapping("/deletar/{id}")
96 @Transactional
97 public String deletarProduto(@PathVariable("id") Long id, Model model){
98     if(!repository.existsById(id)){
99         throw new IllegalArgumentException("Produto inválido - id: " + id);
100     }
101     try {
102         repository.deleteById(id);
103     } catch (Exception e){
104         throw new IllegalArgumentException("Produto inválido - id: " + id);
105     }
106
107     return "redirect:/produtos/listar";
108 }
```

Testar a aplicação

The screenshot shows a web browser window with the address bar at `localhost:8080/produtos/listar`. The page title is "Lista de Produtos". A navigation bar at the top contains "FIAP", "Home", "Cadastrar", and "Produtos". A modal dialog box is open in the center, titled "localhost:8080 diz", with the message "Deseja excluir Smart TV?" and two buttons: "OK" and "Cancelar". Below the dialog, a table lists products:

Nome	Descrição	Valor	Ações
Mouse Microsoft	Mouse sem fio	250.0	<button>Excluir</button> <button>Editar</button>
Smartphone Samsung Galaxy A54 5G	Samsung Galaxy A54 5G	1799.0	<button>Excluir</button> <button>Editar</button>
Smart TV	Smart TV LG LED 65 polegadas	3999.0	<button>Excluir</button> <button>Editar</button>

Testar a aplicação



Nome	Descrição	Valor	Ações
Sem produtos			

■ Editar View listar-produtos.html

```
listar-produtos.html x
22     </thead>
23     <tbody>
24     <tr th:if="${produtos.empty}">
25         <td colspan="4">Lista de produtos vazia</td>
26     </tr>
27     <tr th:each="produto : ${produtos}" >
28         <td><span th:text="${produto.nome}"> Nome </span></td>
```

I Testar a aplicação

FIAP	Home	Cadastrar	Produtos
Lista de Produtos			
Nome	Descrição	Valor	Ações
Lista de produtos vazia			



Copyright © 2024
Prof^a. Aparecida de Fátima Castello Rosa

Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito, do Professor (autor).