

	<p>Universidade Estácio Campus Polo Barra – RJ Curso de Desenvolvimento Full Stack Relatório da Missão Prática 4 – Mundo 3</p>
Disciplina:	RPG0017 - Vamos integrar sistemas
Nome:	Lucia Maria de Lima Martins – 202309761581
Turma:	9001 – 3º Semestre

Implementação de sistema cadastral com interface Web, baseado nas tecnologias de Servlets, JPA e JEE.

Objetivos da prática

1. Implementar persistência com base em JPA.
2. Implementar regras de negócio na plataforma JEE, através de EJBs.
3. Implementar sistema cadastral Web com base em Servlets e JSPs.
4. Utilizar a biblioteca Bootstrap para melhoria do design.
5. No final do exercício, o aluno terá criado todos os elementos necessários para exibição e entrada de dados na plataforma Java Web, tornando-se capacitado para lidar com contextos reais de aplicação.

1º Procedimento | Camadas de Persistência e Controle

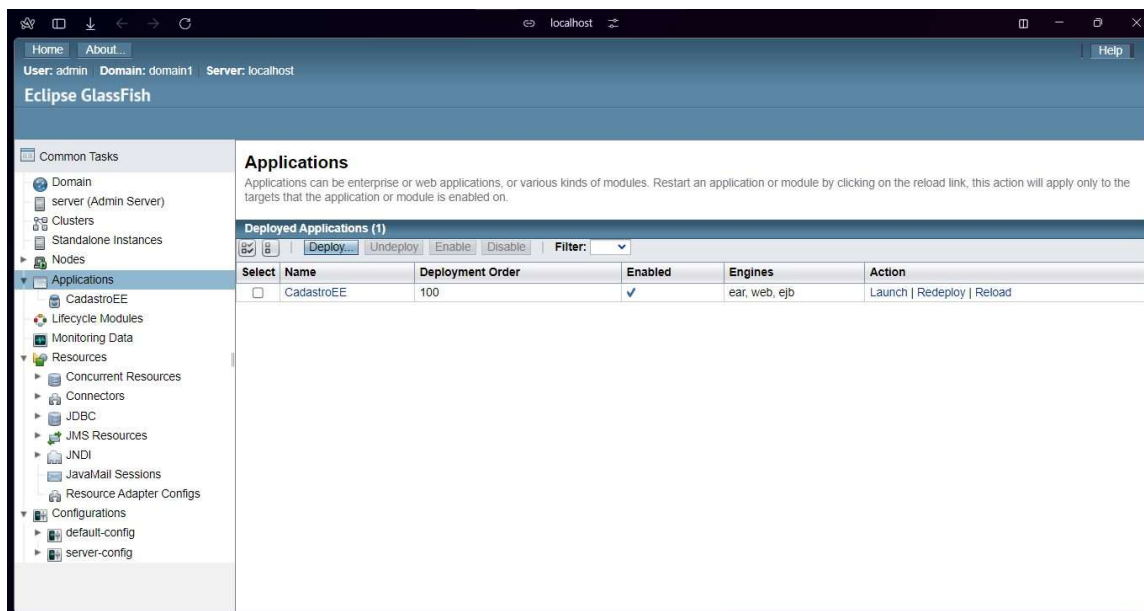


Figura 01: Application rodando no GlassFish server.

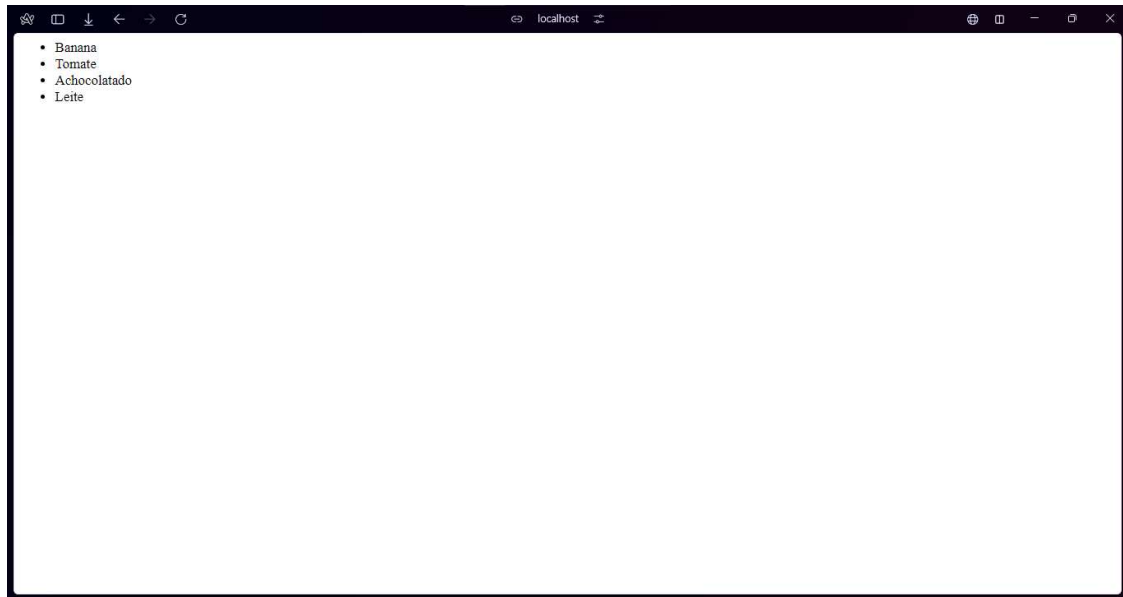


Figura 02: ServletProduto.

Análise e Conclusão:

a) Como é organizado um projeto corporativo no NetBeans?

Um projeto corporativo no NetBeans é organizado em três módulos principais (EAR, EJB, WAR), que correspondem à separação de camadas da aplicação (apresentação, lógica de negócios e persistência). Seguir essa estrutura facilita o desenvolvimento, a manutenção e o deployment da aplicação em ambientes empresariais.

b) Qual o papel das tecnologias JPA e EJB na construção de um aplicativo para a plataforma Web no ambiente Java?

JPA (Java Persistence API) é uma tecnologia que permite que aplicativos Java armazenem e acessem facilmente dados em um banco de dados, ela é uma forma simples e eficiente de trabalhar com informações em redes.

EJB (Enterprise JavaBeans) são usados para o desenvolvimento e implantação de aplicações distribuídas baseadas em componentes que são escaláveis, transacionais, e seguros. Um EJB normalmente contém a lógica de negócio que atua sobre os dados de negócio.

c) Como o NetBeans viabiliza a melhoria de produtividade ao lidar com as tecnologias JPA e EJB?

O NetBeans oferece diversas funcionalidades que facilitam e melhoram a produtividade ao trabalhar com JPA e EJB, como Assistentes e Wizards de Código, Integração com Servidores de Aplicação, Gerenciamento de Persistência e Configuração Simplificada, Suporte a Anotações e Código Automático, Depuração e Teste Integrados, e Suporte a Frameworks e

Extensões. Essas funcionalidades reduzem a quantidade de código manual, minimizam erros e aceleram o processo de desenvolvimento.

- d) O que são Servlets, e como o NetBeans oferece suporte à construção desse tipo de componentes em um projeto Web?

Servlets são classes Java, desenvolvidas de acordo com uma estrutura bem definida que quando instaladas e configuradas em um Servidor que implemente um Servlet Container, podem tratar requisições recebidas de clientes Web. Ao receber uma requisição, um Servlet pode capturar os parâmetros desta requisição, efetuar qualquer processamento inerente a uma classe Java, e devolver uma página HTML.

O NetBeans simplifica a criação, configuração, deploy e depuração desses componentes, com ferramentas integradas como assistentes, suporte a anotações, deploy automático, e depuração robusta, NetBeans melhora a produtividade dos desenvolvedores ao trabalhar com servlets.

- e) Como é feita a comunicação entre os Servlets e os Session Beans do pool de EJBs?

A comunicação entre servlets e session beans (EJBs) é tipicamente realizada através de injeção de dependência com a anotação `@EJB`. Esse processo é gerenciado automaticamente pelo contêiner Java EE, que cuida do ciclo de vida e das transações.

2º Procedimento | Interface Cadastral com Servlet e JSPs

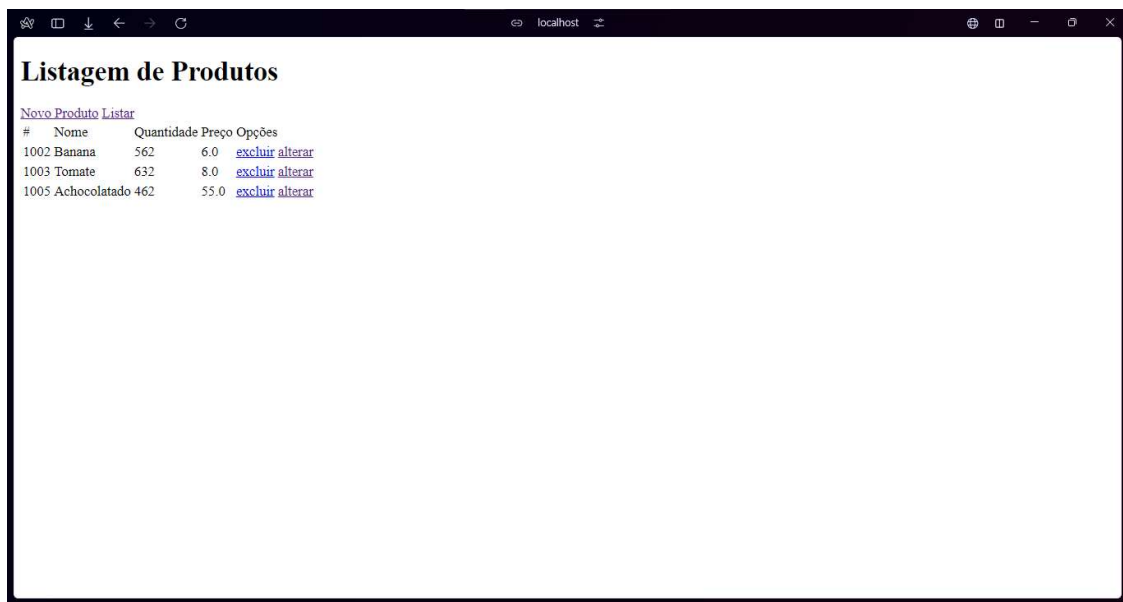


Figura 03: ProdutoLista.jsp.

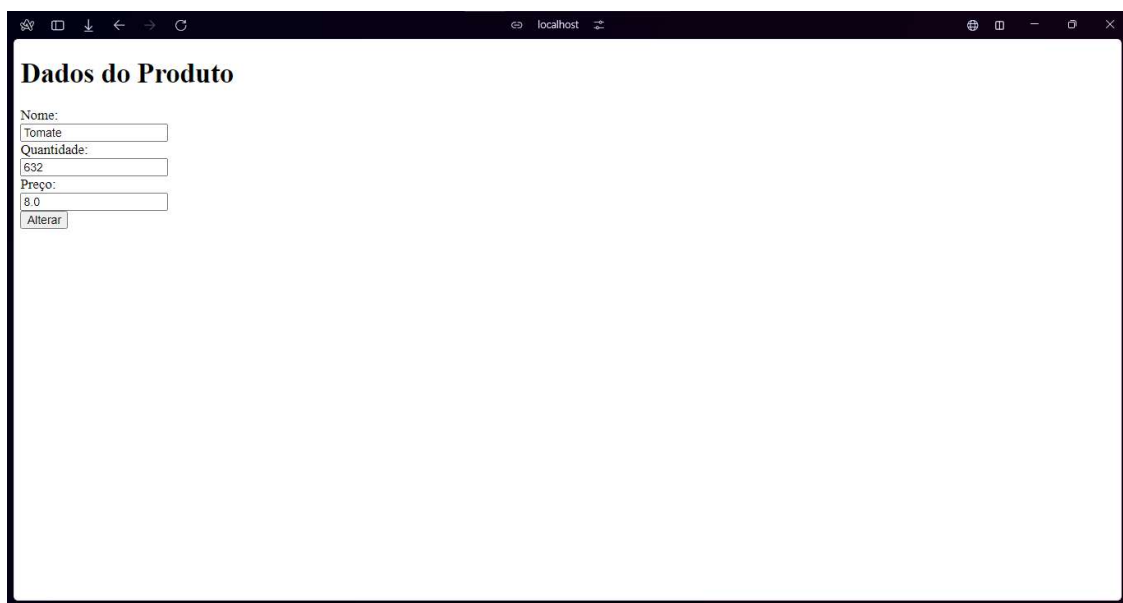


Figura 04: ProdutoDados.jsp.

Análise e Conclusão

- a) Como funciona o padrão Front Controller, e como ele é implementado em um aplicativo Web Java, na arquitetura MVC?

Basicamente o Front Controller trata todas as chamadas vindas de um site web e é organizado em duas partes: através de um Manipulador Web e uma hierarquia de Comandos. O Manipulador Web é o objeto que efetivamente recebe as solicitações HTTP do tipo POST ou GET do servidor web. Ele extrai as informações necessárias da URL e das solicitações e então decide que tipo

de ação iniciar e por fim delega a um objeto Comando para executar a ação. Vale salientar que tanto o Manipulador Web quanto o objeto Comando são partes do Front Controller. Dessa forma, o Comando escolhe qual Visão (ou página) usar para a resposta. O Manipulador Web tem como única responsabilidade escolher qual Comando executar.

O Front Controller é frequentemente utilizado em conjunto com a arquitetura Model-View-Controller (MVC). O MVC é um padrão de arquitetura que separa a lógica de negócio (Model), a apresentação dos dados (View) e o controle das requisições (Controller). O Front Controller atua como o Controller do MVC, recebendo as requisições e direcionando-as para o devido tratamento.

b) Quais as diferenças e semelhanças entre Servlets e JSPs?

Servlets e JSP são tecnologias usadas para criar aplicações web em Java, mas possuem abordagens bem distintas.

Servlets são classes Java que permitem gerar código HTML a partir de instruções e comandos em Java; assim, os servlets são mais adequados para a lógica de negócios e processamento de dados.

Por outro lado, JSP são páginas HTML com capacidade de incorporar código Java; logo são mais adequados para a apresentação da interface com o usuário.

Ambos, servlets e JSP, são executados no servidor de aplicações (GlassFish, Tomcat), com a possibilidade de interagir com bancos de dados, assim como outras tecnologias Java Enterprise.

Em resumo, enquanto servlets são mais voltados para controle, JSPs são mais adequados para a visualização e, frequentemente, ambos são usados em conjunto para separar a lógica de negócios da interface de usuário.

c) Qual a diferença entre um redirecionamento simples e o uso do método forward, a partir do RequestDispatcher? Para que servem parâmetros e atributos nos objetos HttpRequest?

A diferença está entre redirecionar o cliente para uma página (sendRedirect) e encaminhar uma requisição para ser atendida por outro recurso (forward).

No primeiro caso (sendRedirect), o cliente receberá uma resposta http em cujo header haverá a informação de que ele deve requisitar outra página, e o browser fará esta requisição. Ou seja, o redirecionamento ocorre no lado no cliente.

No segundo caso (forward), no lado do server a requisição do usuário será encaminhada para ser atendida por outro recurso (outro servlet). Este outro servlet eventualmente devolverá outra página para o usuário.

Os parâmetros e atributos em objetos HttpRequest são fundamentais para passar informações entre cliente e servidor, ou entre diferentes partes do servidor. Parâmetros são tipicamente usados para enviar dados de formulários ou de solicitações de URL, enquanto atributos são usados para manter dados durante a vida útil de uma requisição ou sessão, de modo a permitir a comunicação entre diferentes componentes do servidor.

3º Procedimento | Melhorando o Design da Interface

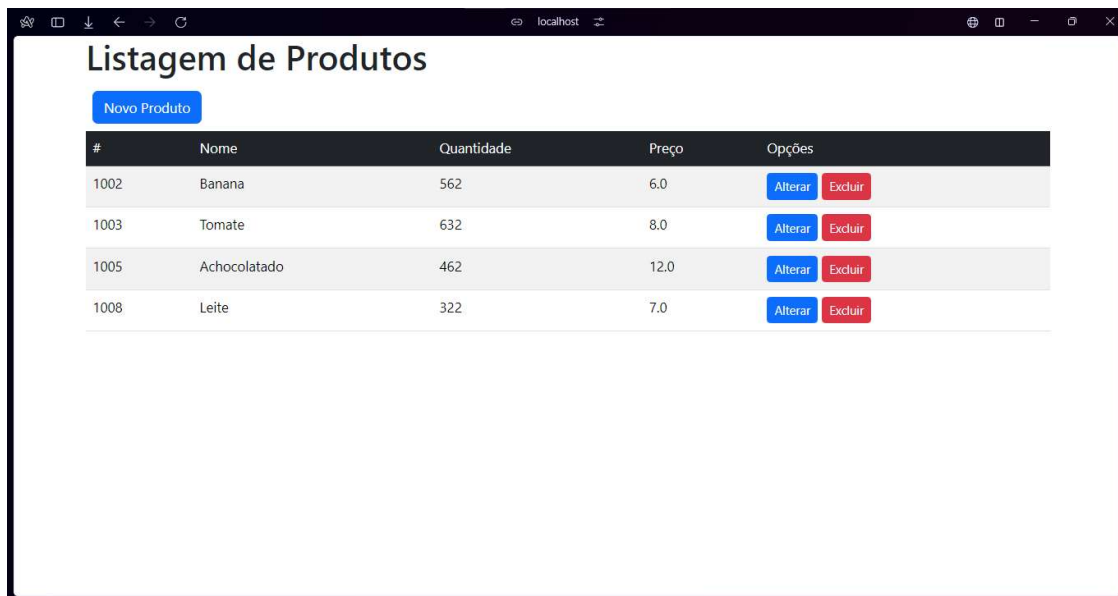


Figura 05: Pagina ProdutoLista estilizada usando Bootstrap.

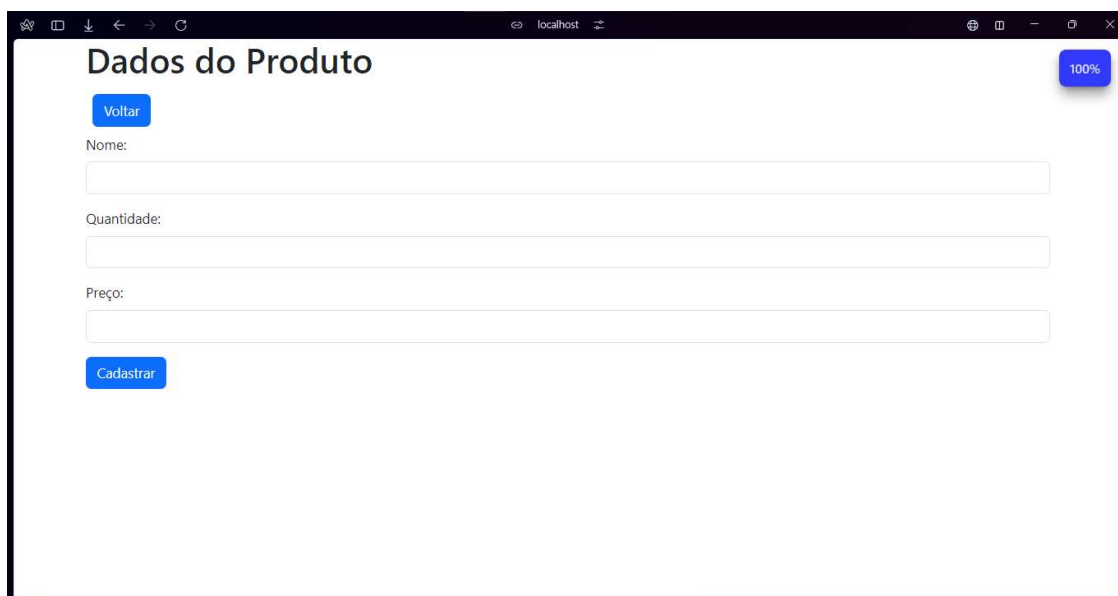
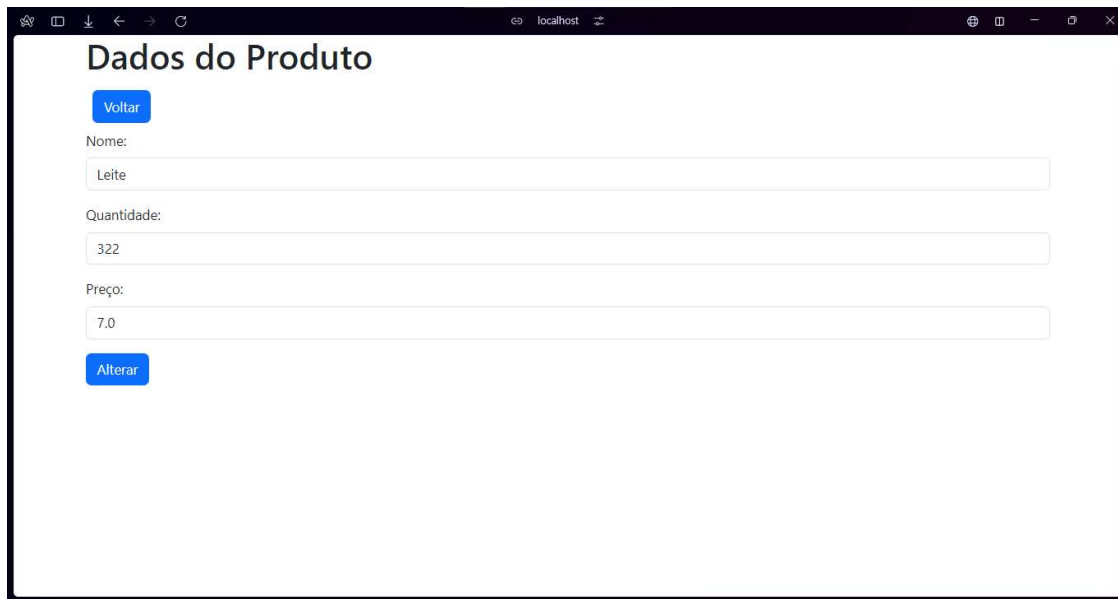


Figura 06: Pagina ProdutoDados estilizada usando Bootstrap.



Dados do Produto

[Voltar](#)

Nome:

Quantidade:

Preço:

[Alterar](#)

Figura 07: Pagina ProdutoDados alterando dados de um Produto.

Análise e Conclusão

a) Como o framework Bootstrap é utilizado?

Bootstrap é um framework front-end usado para desenvolver aplicações web e sites mobile-first, ou seja, com o layout adaptado à tela do dispositivo utilizado pelo usuário.

O Bootstrap é formado por uma série de arquivos CSS e JavaScript que são responsáveis por atribuir características específicas aos elementos da página. Há um arquivo principal no framework, chamado bootstrap.css, que contém a definição para todos os estilos utilizados. Basicamente, a estrutura do framework é formada por dois diretórios:

- **css:** que contém os arquivos necessários para a estilização dos elementos e uma alternativa ao tema original;
- **js:** contém as versões do arquivo bootstrap.js (original e minificado), que é responsável por executar as aplicações de estilo que necessitam de manipulação interativa.

Para atribuir uma característica a um elemento, basta informar a classe correspondente na propriedade class do elemento a ser estilizado. Veja o exemplo:

```

```

Nesse exemplo, foi atribuído o conteúdo “rounded-sm” para a propriedade da imagem. Trata-se de um estilo que adiciona cantos arredondados no

elemento. Portanto, ao carregar a imagem, as características referentes a essa classe serão aplicadas no elemento.

b) Por que o Bootstrap garante a independência estrutural do HTML?

O Bootstrap é um framework gratuito que tem ganhado cada vez mais popularidade entre os desenvolvedores front-end, ele pode ser usado facilmente e poupa muito tempo dos desenvolvedores em não ter mais que escrever manualmente as sintaxes todas as vezes que forem programar.

O framework também é muito flexível e pode acomodar quase todas as necessidades do desenvolvimento web front-end. As suas melhores habilidades incluem, mas não se limitam, a funções que permitem que páginas na internet funcionem em telas de qualquer tamanho.

c) Qual a relação entre o Bootstrap e a responsividade da página?

O sistema grid Bootstrap usa vários containers, linhas e colunas para arranjar e alinhar conteúdo. Seu sistema de grid responsivo, permite criar facilmente layouts de colunas que se reorganizam automaticamente em diferentes tamanhos de tela.

- Os containers são elementos que envolvem o conteúdo do site e ajudam a alinhar corretamente os elementos dentro da página.
- Os containers podem ser do tipo `‘.container’` ou `‘.container-fluid’`, cada um com suas características específicas.
- As linhas são usadas para agrupar colunas horizontalmente e devem estar dentro de um container.
- As colunas são responsáveis por exibir o conteúdo e devem sempre estar dentro de uma linha.