

## 1. Relatório da Missão Prática - Nível 4

UNIVERSIDADE ESTÁCIO DE SÁ  
Tecnologia em Desenvolvimento Full Stack  
RPG0017 - Vamos Integrar Sistemas

Aluno: Carlos Alexandre Paulino de Oliveira  
Matrícula: 202308511361

### Objetivos da Prática

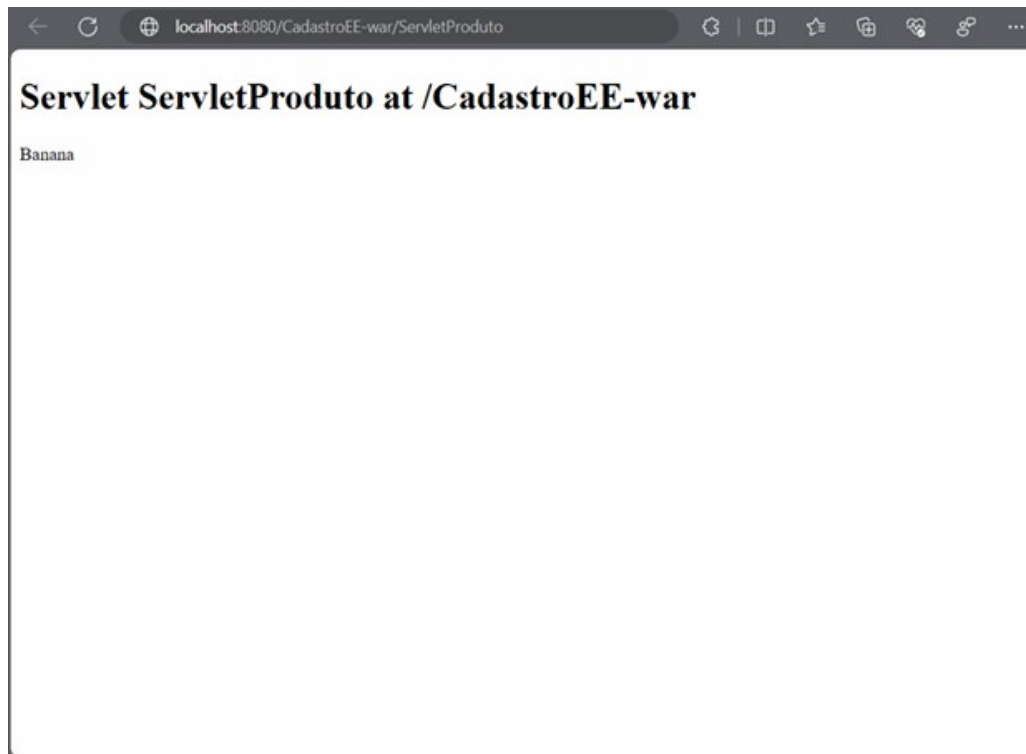
O objetivo principal deste projeto é o desenvolvimento de um sistema cadastral com interface web, utilizando tecnologias como Servlets, JPA e JEE. Este projeto visa proporcionar uma compreensão abrangente sobre a criação de aplicativos web na plataforma Java, abordando desde a implementação da camada de persistência até a criação de uma interface amigável para o usuário.

### Objetivos Específicos:

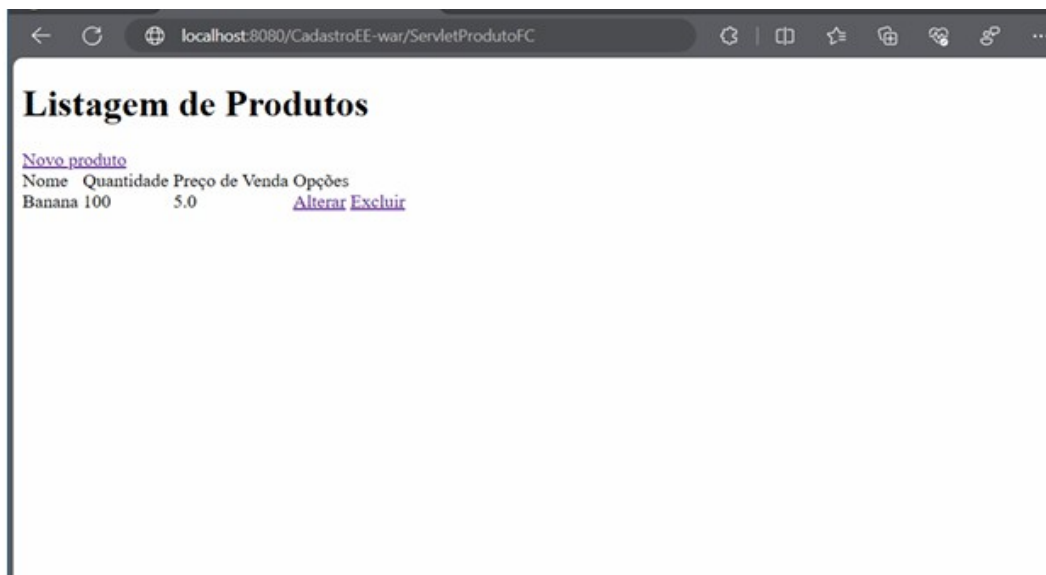
- Implementar a persistência de dados com Java Persistence API (JPA), garantindo a integridade e segurança das informações.
- Desenvolver regras de negócio utilizando a plataforma Java Enterprise Edition (JEE) com o uso de Enterprise JavaBeans (EJBs), otimizando o processamento dos dados.
- Criar uma interface web usando Servlets e JavaServer Pages (JSPs) para facilitar a interação do usuário.
- Aplicar o framework Bootstrap para aprimorar o design e a usabilidade, proporcionando uma experiência agradável e acessível.

## Resultados

1º Procedimento: Desenvolvimento das camadas de persistência e controle.



2º Procedimento: Criação da interface de cadastro utilizando Servlet e JSPs.



← ↻ http://localhost:8080/CadastroEE-war/ServletProdutoFC?acao=incl...

## Dados do Produto

Nome:  Quantidade:  Preço de venda

3º Procedimento: Aperfeiçoamento do design da interface com Bootstrap.

← ↻ http://localhost:8080/CadastroEE-war/ServletProdutoFC

## Listagem de Produtos

Nome	Quantidade	Preço de Venda	Opções
Banana	100	5.0	<input type="button" value="Alterar"/> <input type="button" value="Excluir"/>
Laranja	500	2.0	<input type="button" value="Alterar"/> <input type="button" value="Excluir"/>
Manga	800	4.0	<input type="button" value="Alterar"/> <input type="button" value="Excluir"/>

← ↻ http://localhost:8080/CadastroEE-war/ServletProdutoFC?acao=incl...

## Dados do Produto

Nome:

Quantidade:

Preço de Venda:

## Análise e Conclusão

Organização de um projeto corporativo no NetBeans: Um projeto corporativo no NetBeans é estruturado hierarquicamente, separando camadas e componentes para facilitar o desenvolvimento e a manutenção. Essa estrutura modular inclui um projeto principal, que encapsula subprojetos dependentes, como EJBs e camada web.

Papel das tecnologias JPA e EJB na construção de aplicativos Java Web: A JPA gerencia a persistência de dados, enquanto os EJBs encapsulam a lógica de negócios em um ambiente seguro e transacional. Juntos, eles suportam o desenvolvimento escalável de aplicativos Java seguindo o padrão MVC.

Facilidades do NetBeans para JPA e EJB: O NetBeans melhora a produtividade ao fornecer suporte para servidores Java EE, assistência durante a codificação, geração de código e depuração. Ferramentas de gerenciamento de banco de dados também são integradas, otimizando o desenvolvimento.

Definição de Servlets e suporte do NetBeans: Servlets permitem o processamento de requisições HTTP, gerando respostas dinâmicas. O NetBeans simplifica o desenvolvimento de Servlets por meio de assistentes e modelos.

Comunicação entre Servlets e Session Beans do pool de EJBs: Realizada via injeção de dependência (@EJB) ou lookup com JNDI, esta comunicação permite que Servlets deleguem a lógica de negócios aos EJBs, promovendo a separação de responsabilidades.

Implementação do padrão Front Controller na arquitetura MVC: Centralizando o processamento de requisições, o Front Controller otimiza a organização e manutenção do código em aplicações Java Web, coordenando o fluxo de trabalho para controladores específicos.

Diferenças e semelhanças entre Servlets e JSPs: Ambos são usados para responder a requisições HTTP, mas Servlets processam a lógica de negócios, enquanto JSPs são focadas na apresentação do conteúdo. Servlets oferecem melhor organização, enquanto JSPs facilitam a criação de interfaces dinâmicas.

Diferença entre redirecionamento e forward via RequestDispatcher: O redirecionamento faz o cliente emitir uma nova solicitação, enquanto o forward processa a resposta internamente. Parâmetros e atributos ajudam na troca de dados durante o ciclo de requisição-resposta.

Uso do Bootstrap: Bootstrap permite criar interfaces responsivas e atraentes, com componentes pré-estilizados que facilitam o desenvolvimento de layouts web consistentes e intuitivos.

Independência estrutural do HTML com Bootstrap: A separação entre HTML e estilo visual permite um código limpo, modular e fácil de manter, além de garantir uma interface responsiva sem alterar a estrutura HTML subjacente.

Bootstrap e a responsividade da página: Com seu sistema de grid flexível, o Bootstrap permite que layouts se adaptem automaticamente a diferentes dispositivos, garantindo uma experiência uniforme para os usuários.