

Disciplina:	RPG0017 – Vamos Integrar sistemas
Nome:	Iggor Motta de Oliveira
Turma:	2024.3
Campus:	EAD
Curso:	Desenvolvimento Full stack
Link:	https://github.com/Iggor-motta/Miss-oPr-ticaN-vel04Mundo3.git

RPG0017 – Vamos integrar sistemas

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

Objetivo da Prática:

- Implementar persistência com base em JPA.
- Implementar regras de negócio na plataforma JEE, através de EJBs.
- Implementar sistema cadastral Web com base em Servlets e JSPs.
- Utilizar a biblioteca Bootstrap para melhoria do design.
- Tornando o Aluno capaz de lidar contexto reais de aplicação.

1º - Procedimento Camadas de Persistência e Controle:

Perguntas e respostas.

Análise e Conclusão:

a) Como é organizado um projeto corporativo no NetBeans?

- Em Módulos e Pacotes, Camadas da arquitetura. Projeto principal encapsula essas camadas, permitindo uma gestão modular e escalável com a camada apresentação (WAR), e a persistência(JPA).

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

- O JPA (Java Persistence API) simplifica o mapeamento objeto-relacional, facilitando a persistência de dados em bancos de dados relacionais. O EJB (Enterprise JavaBeans) fornece componentes de negócios, como os Session Beans, que encapsulam a lógica de negócios e podem ser gerenciados pelo contêiner Java EE, permitindo transações e segurança de maneira declarativa.

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

- O NetBeans oferece ferramentas visuais para gerar entidades JPA diretamente a partir do banco de dados e para criar Session Beans automaticamente. Isso reduz o tempo de desenvolvimento e minimiza erros manuais, aumentando a produtividade.

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 que processam requisições HTTP em aplicativos Web. O NetBeans facilita a criação e o gerenciamento de Servlets através de assistentes de configuração, suporte à anotação e integração com descritores de implantação (web.xml).

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

- A comunicação entre Servlets e Session Beans é realizada via injeção de dependência, utilizando a anotação @EJB. Isso permite que o Servlet invoque métodos nos Session Beans para acessar a lógica de negócios e interagir com o banco de dados de forma simplificada e segura.

2º - Procedimento Interface Cadastral com Servlet e JSPs :

Perguntas e respostas.

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

- O padrão Front Controller centraliza o processamento das requisições em um único ponto de entrada, neste caso, o ServletProdutoFC. Ele decide qual ação executar e qual página JSP deve ser carregada com base nos parâmetros recebidos. Esse padrão é fundamental na arquitetura MVC, pois separa a lógica de controle da lógica de apresentação.

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

- Servlets: São componentes Java que respondem a requisições HTTP e são mais adequados para processamento de lógica de negócios e controle de fluxo.
JSPs: São páginas HTML dinâmicas que incorporam código Java, sendo mais adequadas para a geração de conteúdo dinâmico, como formulários e tabelas.
- Semelhanças: Ambos podem manipular requisições HTTP e trabalhar com o objeto HttpServletRequest para acessar parâmetros e atributos
- Diferenças: A principal diferença está na sua função: Servlets focam no controle, enquanto JSPs focam na apresentação

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?

- Redirecionamento Simples (sendRedirect): O navegador do cliente é redirecionado para uma nova URL, causando uma nova requisição HTTP.
- Método forward: O RequestDispatcher.forward encaminha a requisição para outro recurso no servidor (como uma JSP) sem mudar a URL do navegador, permitindo o compartilhamento de atributos entre o Servlet e a JSP.

- Parâmetros e Atributos no HttpServletRequest: Parâmetros são dados enviados pelo cliente (e.g., via query string ou formulário), enquanto atributos são dados que o servidor pode armazenar e passar entre Servlets e JSPs durante o processamento da mesma requisição.

3º - Procedimento Melhorando o Design da Interface:

Perguntas e respostas.

a) Como o framework Bootstrap é utilizado?

- O Bootstrap foi utilizado para estilizar as páginas JSP de forma eficiente, aplicando classes CSS pré-definidas que proporcionam um design moderno e responsivo sem a necessidade de escrever código CSS manualmente. A inclusão foi feita via CDN, o que simplifica a manutenção e atualização do framework.

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

- O Bootstrap garante a independência estrutural do HTML ao permitir que o desenvolvedor aplique estilos e layouts complexos através de classes CSS, sem alterar a estrutura HTML subjacente. Isso facilita a reutilização de componentes e a manutenção do código.

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

- A relação entre o Bootstrap e a responsividade da página é direta. O Bootstrap oferece uma grade flexível e classes responsivas que se adaptam automaticamente a diferentes tamanhos de tela, garantindo que o conteúdo seja exibido corretamente em dispositivos móveis, tablets e desktops.

Códigos solicitados na missão encontra-se neste link

<https://github.com/Iggor-motta/Miss-oPr-ticaN-vel04Mundo3.git>