



Matheus Bicalho Costa Lemberg – [202208610773](#)

Polo Santa Inês – Belo Horizonte – MG

Turma 2022.3

Vamos integrar sistemas

Objetivos 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 JPS.

Utilizar a biblioteca bootstrap para melhoria do design.

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.

Análise e conclusão

Como é organizado um projeto corporativo no Netbeans?

No NetBeans, um projeto corporativo é estruturado conforme o tipo de aplicativo desejado, como uma aplicação Java EE ou Web. Após a criação, o NetBeans organiza os arquivos em uma hierarquia de diretórios específica para cada tipo de projeto. O "Navegador de Projetos" exibe essa estrutura, mostrando pastas para diferentes componentes, como páginas web, controladores, recursos estáticos e configurações. Os pacotes Java são organizados de acordo com a arquitetura do aplicativo para melhorar a manutenção. Além disso, o NetBeans oferece ferramentas para gerenciar bibliotecas, configurar opções de compilação e implantação, facilitando o desenvolvimento e a administração do projeto corporativo.

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

A Java Persistence API (JPA) mapeia objetos Java em bancos de dados, simplificando a persistência de dados. Enterprise JavaBeans (EJBs) gerenciam lógica de negócios e transações, permitindo reutilização de código e gestão eficiente de operações em aplicativos corporativos na web. Juntos, oferecem persistência e operações transacionais robustas.

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

O NetBeans agrega produtividade ao manipular as tecnologias JPA e EJB através de ferramentas integradas. Com suporte embutido para a JPA, oferece assistentes de criação de entidades e

mapeamento, agilizando o desenvolvimento do acesso a dados. Além disso, proporciona geração automática de código para simplificar a criação de EJBs, gerenciando aspectos como transações e segurança. A IDE disponibiliza ferramentas de depuração e perfis para otimizar o desempenho de aplicativos que utilizam essas tecnologias, enquanto o "Navegador de Projetos" facilita a organização e navegação dentro dos projetos, viabilizando uma integração fluída com JPA e EJB, possibilitando um ambiente eficiente para o desenvolvimento de aplicativos corporativos na plataforma Java.

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

Servlets são componentes Java utilizados para criar aplicativos web dinâmicos. Eles processam solicitações do cliente, como HTTP, realizando a lógica de negócios e gerando respostas dinâmicas. O NetBeans oferece suporte à construção de Servlets através de assistentes e modelos de código, simplificando sua criação. A IDE disponibiliza ferramentas para criação, edição e depuração de Servlets, além de facilitar a integração com outros componentes web, como JSPs (JavaServer Pages) e bibliotecas para desenvolvimento eficiente de aplicativos web baseados em Servlets.

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

O padrão Front Controller é usado em aplicações web Java com a arquitetura MVC para gerenciar centralmente o fluxo de requisições. O Front Controller recebe todas as solicitações e as direciona para controladores específicos com base em critérios como URL. Os controladores, responsáveis pela lógica de negócios, interagem com o modelo e atualizam as visualizações. Esse padrão simplifica o roteamento de requisições, promove a organização modular do código e mantém a consistência no tratamento das solicitações.

Quais as diferenças e semelhanças entre Servlets e JSPs?

Diferenças:

- Servlets: Componentes Java para controle e processamento de requisições HTTP.
- JSPs: Arquivos que combinam HTML e código Java para páginas dinâmicas.

Semelhanças:

- Ambos criam aplicações web no ambiente Java.
- Executados no servidor, interagem com o cliente via navegador.
- Acessam recursos do Java EE, como bancos de dados e Session Beans.
- Geralmente, são integrados, com Servlets controlando a lógica e direcionando para JSPs na apresentação.

Como é organizado um projeto corporativo no Netbeans?

Um projeto corporativo Java EE é uma estrutura organizacional composta por diversos módulos para desenvolver aplicativos empresariais de alta qualidade: Módulo EJB – contém componentes para gerenciar lógica de negócios, transações e operações complexas; Projeto WEB – inclui recursos de interface de usuário, como páginas web, para interação em aplicativos Java EE; Módulo JAR – armazena classes reutilizáveis compartilhadas entre partes do aplicativo ou outros aplicativos Java EE; Diretórios de configuração – mantém arquivos como “web.xml” e “META-INF” para configurar comportamentos específicos do aplicativo; Servidores de aplicativos Java EE - plataformas que hospedam e gerenciam aplicativos, oferecendo recursos como segurança e transações; Gerenciamento de dependências – organiza bibliotecas externas e frameworks essenciais para garantir versões corretas durante o desenvolvimento e implantação.

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

A JPA lida com a camada de persistência simplificando a interação com o banco de dados, enquanto os EJBs são usados para implementar a lógica de negócios em aplicações corporativas Java EE. A JPA e os EJBs trabalham juntos e se complementam nas construções de aplicativos web robustos e escaláveis.

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

O NetBeans oferece melhorias significativas na produtividade ao lidar com as tecnologias JPA e EJB. Com ferramentas integradas, simplifica o desenvolvimento e mapeamento de entidades JPA, proporcionando assistentes para criação de classes de entidades, facilitando a persistência de dados. No caso dos EJBs, o NetBeans agiliza a criação desses componentes, oferecendo suporte para geração automática de código e assistentes de configuração, otimizando a implementação de lógica de negócios distribuída e transacional. A IDE fornece recursos de depuração, perfis de desempenho e integração com servidores de aplicativos, agilizando o ciclo de desenvolvimento. Com isso, o NetBeans simplifica a construção de aplicativos corporativos, garantindo maior eficiência no uso das tecnologias JPA e EJB, permitindo aos desenvolvedores focarem na lógica de negócios e na interação com os dados, aumentando a produtividade e qualidade no desenvolvimento de aplicações Java EE.

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 atuam como controladores em aplicações web, processando solicitações HTTP e gerando respostas dinâmicas. O NetBeans suporta a construção de Servlets através de assistentes de criação de arquivos Servlet, simplificando sua implementação e

fornecendo recursos para edição, depuração e gerenciamento de ciclo de vida. A IDE oferece templates e ferramentas para criar, configurar e integrar Servlets em projetos web, permitindo aos desenvolvedores criar e gerenciar esses componentes de forma eficiente, garantindo a funcionalidade adequada e facilitando a interação entre o cliente e o servidor na plataforma Java EE.

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

A comunicação entre Servlets e Session Beans do pool de EJBs ocorre por meio de invocações de métodos remotos ou locais. Os Servlets, componentes web, obtêm acesso aos Session Beans por meio de recursos de injeção de dependência ou pela obtenção de referências através do contexto Java EE. Essa comunicação se concretiza através da invocação de métodos nos Session Beans, permitindo que os Servlets acessem serviços e funcionalidades encapsuladas pelos EJBs. Dessa maneira, os Session Beans disponibilizam lógica de negócios e operações transacionais para os Servlets, possibilitando uma arquitetura modular e distribuída para o desenvolvimento de aplicações web robustas na plataforma Java EE, facilitando a interação entre os diferentes componentes no ambiente corporativo.

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

O padrão Front Controller é usado em aplicações web Java com a arquitetura MVC para gerenciar centralmente o fluxo de requisições. O Front Controller recebe todas as solicitações e as direciona para controladores específicos com base em critérios como URL. Os controladores, responsáveis pela lógica de negócios, interagem com o modelo e atualizam as visualizações. Esse padrão simplifica o roteamento de requisições, promove a organização modular do código e mantém a consistência no tratamento das solicitações.

Quais as diferenças e semelhanças entre Servlets e JSPs?

Diferenças:

Servlets – são classes Java que respondem a requisições HTTP

JSPs – são páginas HTML com código Java embutido, usados para gerar conteúdo dinâmico e intercalar código Java com marcação HTML.

Semelhanças:

Ambos – são tecnologias Java para desenvolvimento web

Integração – JSPs frequentemente usam servlets nos bastidores para processar solicitações

MVC – podem ser utilizados juntos seguindo o padrão MVC para separar lógica de apresentação e controle.

Qual a diferença entre um redirecionamento simples e o uso do método forward a partir do RequestDispatcher?

Um redirecionamento simples realiza uma nova solicitação ao cliente, enquanto o método `forward` do `RequestDispatcher` encaminha internamente a requisição para outro recurso dentro do servidor. No redirecionamento simples, o cliente faz uma nova requisição ao recurso direcionado. Já o `forward`, realizado no lado do servidor, transfere o controle da requisição para outro recurso (página, `Servlet`, etc.) no mesmo ciclo de requisição, mantendo os dados da requisição original, garantindo processamento interno e evitando uma nova solicitação do cliente, útil para encadear ações em um mesmo contexto de requisição.

Para que servem os parâmetros e atributos nos objetos `HttpRequest`?

Os parâmetros são dados enviados com a solicitação HTTP e geralmente usados para transmitir informações na URL. Acessíveis através dos métodos `"getParameter"` e similares. Enquanto os atributos são objetos armazenados no contexto da solicitação e usados para compartilhar dados entre componentes da aplicação. Acessíveis através dos métodos `"setAttribute"` e `"getAttribute"`.

Como o framework Bootstrap é utilizado?

O Bootstrap é utilizado para criar interfaces web elegantes e responsivas, oferecendo uma gama diversificada de ferramentas e componentes que aceleram o desenvolvimento front end e garantem uma experiência consistente do usuário.

Por que o Bootstrap garante a independência estrutural do HTML?

Porque disponibiliza uma variedade de estilos e componentes que dissociam a aparência visual do HTML, permitindo ao desenvolvedor preservar a estrutura semântica do HTML. Dessa forma, é possível aplicar estilos e funcionalidades uniformes por meio das classes e elementos oferecidos pelo Bootstrap, facilitando a personalização e a consistência na apresentação dos elementos na interface web.

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

O Bootstrap facilita a criação de páginas web responsivas por meio do seu sistema de grid e componentes flexíveis, adaptando-se automaticamente a diferentes tamanhos de tela. Isso é alcançado através de classes e estilos predefinidos que ajustam o layout e o design conforme necessário.

