



Missão Prática | Nível 4 | Mundo 3

Filipe Alvim Santos - 202208291325

Campus Polo Sulacap – RJ / Desenvolvimento Full Stack
Nível 4: Vamos Integrar Sistemas – Turma: 22.3 – 3º Semestre

Objetivo da Prática

Os objetivos da prática são: 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.

No final do exercício, o aluno terá criado todos os elementos necessários para a exibição e entrada de dados na plataforma Java Web, tornando-se capacitado para lidar com contextos reais de aplicação.

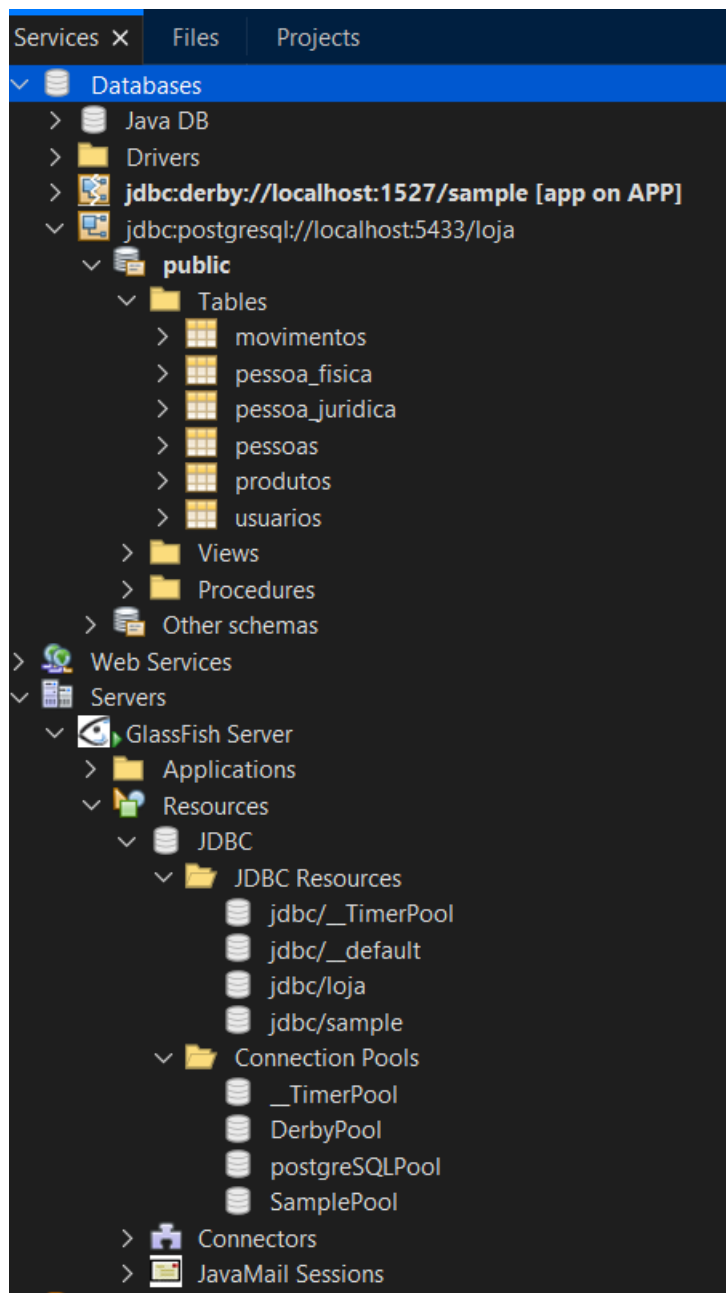
1º Procedimento | Camadas de Persistência e Controle

- Criando a conexão pool e jdbc resource:

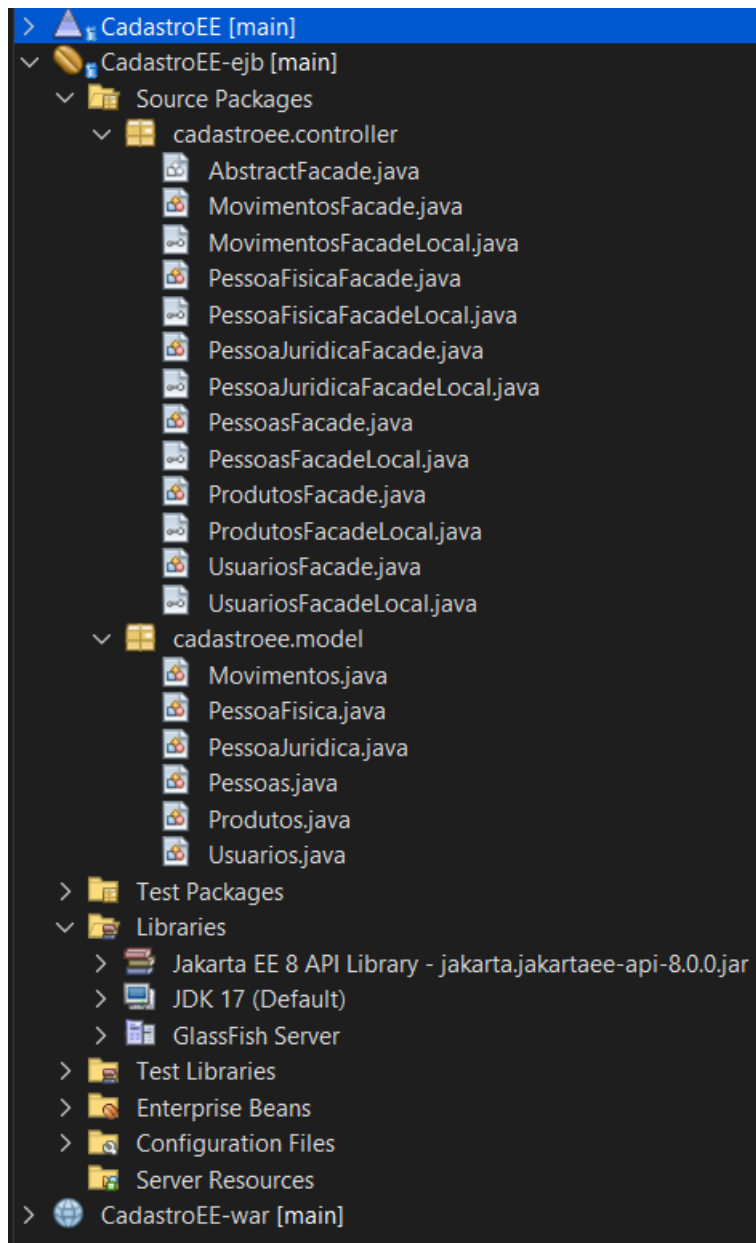
```
create-jdbc-connection-pool
--datasourceclassname org.postgresql.ds.PGSimpleDataSource
--restype javax.sql.DataSource
--property
driverClass=org.postgresql.
Driver:portNumber=5433:password=loja:user=loja:serverName=localhost:databaseName=Loja:URL="jdbc\\:postgresql\\://
localhost\\:5433\\/loja" postgresQLPool

ping-connection-pool postgresQLPool

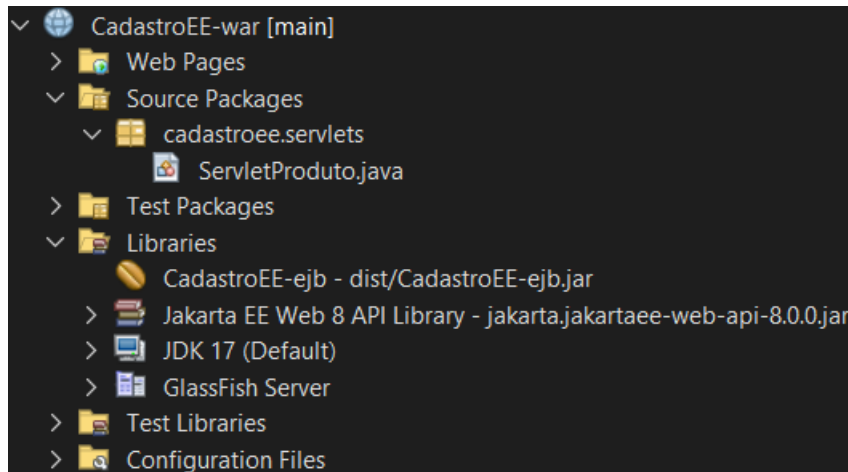
create-jdbc-resource --connectionpoolid postgresQLPool jdbc/loja
```



- Criando Entity classes e session beans:



-Criando ServletProduto:



a) Como é organizado um projeto corporativo no NetBeans?

Resposta: Um projeto corporativo no NetBeans é organizado em módulos EJB e WEB, que podem incluir tecnologias como JSP, Servlet, EJB e JPA.

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

Resposta: A Java Persistence API (JPA) é um framework leve, baseado em POJOs (Plain Old Java Objects), para persistir objetos Java. Ela não é apenas um framework para Mapeamento Objeto-Relacional (ORM - Object-Relational Mapping), mas também oferece diversas funcionalidades essenciais em qualquer aplicação corporativa. A JPA é a solução ORM padrão para as plataformas Java SE e Java EE. Já os Enterprise Java Beans (EJBs) são usados para desenvolver componentes de software e encapsular a lógica do negócio. Eles são usados pelos componentes da camada de apresentação que fornecem essa funcionalidade para os usuários do aplicativo.

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

Resposta: Quando se trata de JPA, o NetBeans permite que você enxergue o conjunto de entidades existentes e os seus relacionamentos, gerenciando todo o acesso ao banco de dados. Isso facilita o processo de configuração e também gera código, permitindo que o desenvolvedor se preocupe mais com a lógica de negócio do que está sendo desenvolvido e menos com implementações secundárias. Em relação ao EJB, o NetBeans também oferece suporte para facilitar o desenvolvimento, permitindo que o desenvolvedor se concentre mais na lógica de negócio.

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

Resposta: Os Servlets são classes Java que, quando instaladas e configuradas em um servidor que implemente um Servlet Container, podem tratar requisições recebidas de clientes Web, como por exemplo os Browsers. Eles são basicamente módulos de software que são executados em um servidor web para atender as requisições de aplicações cliente e prestar-lhes algum tipo de serviço. O NetBeans é capaz de gerar código e arquivos de configuração, auxiliar no empacotamento e instalação de aplicações, renomear elementos do código, entre outras. Tudo isto através de assistentes, que dão o suporte necessário que o desenvolvedor precisa para se preocupar mais com a lógica de negócio e menos com implementações secundárias.

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

Resposta: A comunicação entre os Servlets e os Session Beans do pool de EJBs é feita através de chamadas feitas por um cliente remoto ou cliente local a um componente EJB. Até a versão 3 do EJB, todos os beans de sessão precisavam de uma interface (Local ou Remota) para fazerem o acesso ao componente. Neste cenário, os Servlets atuam como clientes que fazem chamadas aos Session Beans para executar a lógica de negócios.

2º Procedimento | Interface Cadastral com Servlets e JSPs

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

Resposta: O padrão Front Controller é um padrão arquitetural que se comporta como um controlador, tratando todas as solicitações para um site Web e, em seguida, roteando para uma ação (ou comando). Ele é implementado em diversos frameworks de ponta que são utilizados em projetos de diferentes complexidades. Em um aplicativo Web Java, na arquitetura MVC, o Front Controller é muito utilizado. Ele define a centralização das chamadas HTTP e o direcionamento do fluxo de execução do sistema para a visualização correta. A requisição é recebida no Servlet, onde são feitas conversões e chamadas para métodos de negócio, e no final um RequestDispatcher redireciona o fluxo para a página JSP.

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

Resposta: O As semelhanças incluem o fato de que ambos são usados para criar aplicações web dinâmicas e são executados em um contêiner web. Além disso, ambos podem processar requisições HTTP e gerar respostas. As diferenças estão principalmente na maneira como são usados e na sua complexidade. Servlets são classes Java que processam requisições HTTP e geram respostas. Eles são escritos em Java puro e são mais complexos do que JSPs. Por outro lado, JSPs permitem que os desenvolvedores misturem Java com HTML ou XML, tornando a criação de páginas web mais simples e intuitiva. Além disso, a compilação dos Servlets é responsabilidade do programador, enquanto os arquivos JSP são compilados pelo Servlet Container no momento da sua primeira utilização.

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?

Resposta: A diferença entre um redirecionamento simples e o uso do método forward do RequestDispatcher está em como a solicitação é tratada. No redirecionamento, o cliente recebe uma resposta HTTP indicando que ele deve requisitar outra página, e o navegador faz essa nova requisição. Isso é visível para o cliente, pois a URL muda. Já no método forward, a requisição do usuário é encaminhada internamente no servidor para ser atendida por outro recurso, como outro servlet. Isso é invisível para o cliente, pois a URL não muda. Parâmetros e atributos nos objetos HttpRequest servem para passar informações durante o ciclo de vida de uma solicitação. Parâmetros são geralmente usados para enviar dados do cliente para o servidor. Atributos, por outro lado, são usados para manter dados entre as solicitações e são úteis para compartilhar informações entre diferentes partes do aplicativo.

3º Procedimento | Melhorando o Design da Interface

a) Como o framework Bootstrap é utilizado?

Resposta: Para utilizar os recursos do Bootstrap, basta referenciar seus arquivos principais na página principal e aplicar os estilos correspondentes nos elementos da página através de classes específicas. Ele se adapta bem a diferentes tipos de aplicações desenvolvidas em diferentes linguagens de programação e frameworks.

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

Resposta: O Bootstrap garante a independência estrutural do HTML porque ele é um framework CSS que organiza e gerencia o layout de um site. Enquanto o HTML lida com o conteúdo e a estrutura de uma página web, o CSS, no caso o Bootstrap, lida com o layout. Isso significa que o Bootstrap permite que os desenvolvedores estilizem e adicionem funcionalidades às páginas web sem alterar a estrutura HTML. Além disso, o Bootstrap contém componentes internos funcionais e bonitos que são fáceis de personalizar, permitindo que os desenvolvedores criem interfaces de usuário atraentes e responsivas com menos esforço e sem comprometer a estrutura HTML.

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

Resposta: A principal característica do Bootstrap é a responsividade do site, ou seja, seu objetivo é permitir que os elementos da página sejam readaptados para o acesso em diferentes dispositivos, como notebooks, tablets, smartphones e, até mesmo, para monitores maiores que os tradicionais. Ele é prioritariamente orientado ao mobile, o que significa que o desenvolvimento parte de um dispositivo móvel e a partir daí é adicionado um comportamento responsivo para dispositivos com resoluções e dimensões maiores. Isso faz com que o resultado visual para telas menores seja mais adequado do que o processo inverso. Portanto, a relação entre o Bootstrap e a responsividade da página é que o Bootstrap fornece as ferramentas necessárias para criar um design responsivo.