



RPG0017 - Vamos integrar sistemas

Kervini Ribeiro da Silva - 202301206073

Polo Centro – Araraquara – SP

Vamos integrar sistemas – 9001 – 3º Semestre

Objetivo da Prática

Criar um Enterprise Application usando o servidor glassfish para hospedar a aplicação. Aplicar o padrão MVC, fazendo uso das tecnologias JPA e EJB para acesso a dados e logica de negócio, e o uso de Servlets e JSP pages para desenvolver a camada de apresentação do sistema.

GITHUB: <https://github.com/Kervini/Mundo3-Nivel4.git>

1º Procedimento | Camadas de Persistência e Controle

Análise e Conclusão:

a) Como é organizado um projeto corporativo no NetBeans?

R: Os projetos corporativos são divididos em três camadas ou projetos. A primeira delas é a camada de aplicação que encapsula as outras duas camadas, onde tem a responsabilidade de integrar as outras duas camadas, que são projetos dependentes um do outro, mas são projetos diferentes. Dentre as duas restantes temos a camada de EJB, onde fica toda a parte de regra de negócio e acesso ao banco de dados, onde usamos as classes Entity e os Session Beans e distribuimos os objetos pelas demais camadas do sistema, servindo como um recurso de escalabilidade. Já a terceira camada é a camada WEB, onde definimos a apresentação do projeto e suas peculiaridades, com o uso de Servlets, JSP Pages etc.

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

R: Em ambientes corporativos a alta disponibilidade dos aplicativos é algo crucial, por isso, é bastante útil o uso de EJBs que permitem desenvolver a lógica no lado do servidor. Além de serem objetos distribuídos que são compartilhados em diferentes computadores através da rede, possibilitando a escalabilidade horizontal.

O JPA é um framework para persistência de dados que simplifica a comunicação da aplicação com o banco de dados abstraindo a complexidade, além de padronizar o processo tornando-o simples e de fácil manutenção. No JPA o ORM é feito definindo classes Entity, que representam tabelas no banco de dados e usamos anotações para fazer o relacionamento entre objeto e tabela.

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

R: O NetBeans gera classes Entity automaticamente facilitando o uso da tecnologia JPA, a única coisa necessária é uma conexão com o banco de dados, onde é feito o mapeamento tabelas e simplesmente selecionamos as tabelas que desejamos mapear em Entity e assim

o NetBeans faz a criação com todos os recursos necessários para persistência. Em relação a tecnologia EJB o NetBeans nos permite gerar os Session Beans a partir das Entity existentes no projeto e assim a camada de negócio é gerada automaticamente. Esses recursos gerados automaticamente nos geram um ganho muito alto de produtividade em um mercado que a cada dia demanda mais do programador.

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

R: Servlet é uma tecnologia da plataforma JEE, que lida com requisições do servidor web. Os servlets recebem requisições, fazem o tratamento necessário, podendo usar Beans no processo e devolvem resposta para quem solicitou. Servlets são facilmente criados e configurados pela IDE NetBeans.

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

R: O Servlet pode usar um objeto EJB definido na camada de EJB. Esse uso é feito definindo um atributo do Session Bean que desejamos usar e marcá-lo com a anotação `@EJB`, toda a complexidade para comunicação é abstraída pela aplicação.

2º Procedimento | Interface Cadastral com Servlet e JSPs

Análise e Conclusão:

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

R: O padrão Front Controller funciona como um ponto de entrada único para todas as requisições do sistema, faz um pré-processamento e partir disso encaminha as requisições para o Servlet responsável por processar aquela requisição. E isso em conjunto com o padrão MVC garante uma separação de responsabilidades que facilita a manutenção e escalabilidade.

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

R: Apesar de bastante semelhantes Servlets e JSP tem características bastante específicas. Uma página JSP mesmo podendo executar código Java sua principal utilidade é a parte visual. Já o Servlet tem como característica principal a implementação lógica da camada de visualização e não focado em apresentação de dados, principalmente usado para lidar com requisições HTTP.

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?

R: No redirecionamento simples do método RequestDispatcher a página destino é chamada e iniciada sem informações processadas anteriormente. Já com o método forward, podemos reenviar a requisição, ou seja, relançar a requisição para que seu processamento continue. Com isso podemos usar os parâmetros passados pela requisição além de setar novos atributos que serão devolvidos na resposta.

3º Procedimento | Melhorando o Design da Interface

Análise e Conclusão:

a) Como o framework Bootstrap é utilizado?

R: O Bootstrap é uma biblioteca CSS que adicionamos a nossa página HTML e com isso podemos usar as classes disponibilizadas para estilizar nossa página de forma rápida, tudo que precisamos fazer é indicar a classe desejada no atributo class do HTML.

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

R: Suas classes são amplamente usadas em diversos componentes HTML e funcionam de forma eficiente em qualquer estrutura HTML.

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

R: O bootstrap é uma biblioteca rica em recursos de responsabilidade onde a maioria deles são implícitos como o componente container e pode ser facilmente implementado usando componentes como o grid.

Conclusão

A missão foi importante para introduzir as tecnologias usadas para desenvolver e manter um ambiente WEB através da visão do servidor. Também aprendi maneiras mais fáceis de fazer a persistência de dados do sistema através JPA e EJB, também como distribuir os Beans através da rede e escalar minha aplicação, entre diversos outros ensinamentos valiosos.