



## RPG0017 – 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?** 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?** Integração unificada – o NetBeans oferece uma integração completa e um ambiente unificado para JPA e EJB, simplificando o processo de desenvolvimento; Assistentes e geração de código automático – ferramentas avançadas, como assistentes e geração automática de código, aceleram a criação de entidades JPA e EJBs, proporcionando eficiência no desenvolvimento; Depuração integrada – a depuração integrada no NetBeans facilita a identificação rápida de problemas em aplicativos Java EE,

contribuindo para um desenvolvimento mais robusto; Gerenciamento de implantação simplificada - o NetBeans simplifica o gerenciamento de implantação, permitindo visualizar alterações instantaneamente e agilizando o ciclo de desenvolvimento; Suporte a banco de dados e mapeamento de entidades - Recursos integrados de suporte a bancos de dados e mapeamento de entidades facilitam a persistência de dados, tornando a manipulação de dados mais eficiente; Construção e gerenciamento de projetos otimizados - O processo de construção e gerenciamento de projetos Java EE é otimizado no NetBeans, proporcionando uma experiência de desenvolvimento mais fluída; Integração com servidores de aplicativos - A integração eficiente com servidores de aplicativos simplifica as tarefas de implantação e teste, garantindo uma transição suave do desenvolvimento para o ambiente de produção; Ferramentas de persistência visual - auxiliam na modelagem visual de dados, facilitando a compreensão e manipulação de estruturas complexas; Eficiência aprimorada no desenvolvimento Java EE - o NetBeans contribui para a melhoria da eficiência no desenvolvimento de aplicativos Java EE, proporcionando um conjunto abrangente de ferramentas integradas e recursos especializados.

- **O que são Servlets, e como o Netbeans oferece suporte à construção desse tipo de componentes em um projeto Web?** São componentes Java que estendem a capacidade dos servidores web para gerar conteúdo dinâmico. O Netbeans oferece suporte a construção de servlets: Ao criar um novo projeto web no NetBeans, você pode escolher a opção adequada para a plataforma desejada (por exemplo, Java EE ou Jakarta EE), facilitando a criação de servlets, fornecendo assistentes de criação, realizando o mapeamento automáticos de URLs, oferecendo suporte de anotações Java EE, oferecendo ferramentas robustas de depuração e teste, integrando servidores web, gerando código automaticamente, gerenciando as dependências e bibliotecas necessárias para o desenvolvimento, entre outros.
- **Como é feita a comunicação entre os Servlets e os Session Beans do pool de EJBs?** A colaboração entre Servlets e Session Beans do pool de EJBs é crucial na construção de aplicativos empresariais em Java EE. Esse intercâmbio é simplificado por meio de injeção de dependência, onde Servlets podem acessar Session Beans usando anotações. Os Session Beans, ao fornecer interfaces locais, otimizam o desempenho, e o contêiner EJB administra o ciclo de vida desses beans, garantindo reutilização. Os Servlets têm acesso direto aos métodos dos Session Beans para executar operações de negócios, promovendo eficiência e integridade transacional. Essa abordagem simplifica o desenvolvimento de aplicativos Java EE, permitindo que Servlets e Session Beans colaborem para fornecer funcionalidades empresariais robustas.

## **Análise e conclusão**

- **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?** O redirecionamento simples ocorre no lado do cliente, onde o navegador faz uma nova solicitação para uma URL diferente. Enquanto o método forward ocorre no lado do servidor, onde a solicitação é internamente direcionada para outro recurso sem envolvimento do cliente, mantendo a mesma solicitação HTTP.
- **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”.

## **Análise e conclusão**

- **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 fornece um conjunto de estilos e componentes que separam a apresentação visual do HTML, permitindo que o desenvolvedor mantenha a estrutura semântica do HTML enquanto aplica estilos e funcionalidades consistentes através das classes e elementos fornecidos pelo Bootstrap.
- **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.