

Instituto Federal Goiano - Campus Ceres
Bacharelado em Sistemas de Informação
Prof. Me. Ronneesley Moura Teles

Jonathan
Fernando Maciel
Warley Rodrigues

Introdução a JSF

Sumário

1	Introdução	2
1.1	O JavaSecer Faces possui as seguintes partes	2
1.2	Característica marcante	2
1.3	Principais motivos que levaram JFS ao sucesso	2
2	Principais provedores do projeto JSF	3
2.1	Mojorra	3
2.2	MyFaces	3
2.3	ADF	3
3	Componentes do projeto JSF	4
3.1	RichFaces	4
3.2	PrimeFaces	4
3.3	OpenFaces	5
3.4	IceFaces	5
3.4.1	Automatic Ajax	5
3.4.2	Ajax Push	6
3.4.3	Resource Management	6
3.4.4	Scoped Resource Registry	6
3.4.5	Streamlined Configuration	6
3.4.6	Window Scope	6
3.5	EasyFaces	6
3.6	Gmaps4jsf	6
4	Extensões JSF	7
4.1	PrettyFaces	7
4.2	OmniFaces	7
5	Referências Bibliográfica	8

Introdução a JSF

1 Introdução

JavaServer Faces ou apenas JSF, é um framework que permite a criação de interfaces de usuário web colocando componentes visuais pré-prontos, de forma que o desenvolvedor não se preocupe com Javascript e HTML. Basta adicionarmos os componentes (calendários, tabelas, formulários) e eles serão renderizados e exibidos em formato html. Seu ponto forte é um grande número de componentes e um design muito flexível o que fez com que esse Framework crescesse muito abordando novas tecnologias.

Desenvolvido pela comunidade JCP (Java Community Process), atualmente o JSF é o framework preferido para o desenvolvimento de aplicações Web possuindo um excelente conjunto de funcionalidades para o cenário de desenvolvimento em que é inserido, possibilitando ao programador preocupar-se somente com a lógica de negócio deixando as tarefas básicas e trabalhosas por conta do framework.

1.1 O JavaSecer Faces possui as seguintes partes

- Conjunto de componentes pré-fabricados de Interface de Usuário (IU);
- Modelo de programação orientado a eventos;
- Modelo de componentes que permite a desenvolvedores independentes fornecer componentes adicionais.

1.2 Característica marcante

- Entidades e regras de negócio;
- Objetos gerais da aplicação (dados, etc.) Visão;
- Componentes UI em páginas JSP/XHTML;
- Kits renderizadores (HTML, WML, XML, etc.) Controlador;
- Faces Servlet (Front Controller);
- Backing Bean (Page Controller ou Modelo).

1.3 Principais motivos que levaram JFS ao sucesso

- Ser uma especificação EE (Enterprise Edition) desde da 5ª Versão;
- Ser um framework cuja a API foi pensada no trabalho dos desenvolvedores de IDEs;
- Possuir comunidades ativas em fóruns, assim facilitando a procura sobre dúvidas e sugestões;
- Exigir baixo conhecimento inicial para criação de interfaces de usuários tradicionais;

- Possuir ótimas bibliotecas de componentes livres e pagas desenvolvidas por terceiros;
- Possuir Ajax nativo em sua versão 2.0.

A aplicação Desktop é um tipo de aplicação que é instalada no computador local e acessa o banco de dados ou gerenciador de arquivos, para o uso de sua aplicação é utilizado Delphi, VB (Visual Basic) ou, no mundo Java, Swing. Ela possui componentes para cada tipo de sistema um desenvolvimento rico possuindo calendários, menus diversos ou componentes drag and drop (arrastar e soltar).

2 Principais provedores do projeto JSF

2.1 Mojarra

A tecnologia JavaServer Faces simplifica a criação de interfaces de usuário para aplicativos JavaServer. Os desenvolvedores podem criar aplicativos da Web montando componentes de UI reutilizáveis em uma página, conectando esses componentes a uma fonte de dados do aplicativo e criação de eventos gerados pelo cliente para manipulando os eventos do lado do servidor. Este projeto fornece informações sobre o desenvolvimento contínuo da especificação JavaServer Faces.

2.2 MyFaces

O Apache MyFaces é um projeto da Apache Software Foundation e hospeda vários subprojetos relacionados à tecnologia JavaServer fornecendo tecnologias como:

- Implementação JavaServer (MyFaces Core, providing api/impl and bundle modules);
- Várias bibliotecas de componentes que contêm widgets UI para construir web-applications com JSF (MyFaces Tomahawk, MyFaces Trinidad, MyFaces Tobago);
- Patoces de extensão para JavaServer (MyFaces Orchestra, MyFaces Extensions Validator, MyFaces Extensions CDI);
- Módulos de integração para outras tecnologias e padrões (MyFaces Portlet Bridge for integration with the portlet-standard).

2.3 ADF

O Oracle ADF é uma estrutura de Java EE de ponta a ponta que simplifica o desenvolvimento de aplicativos, fornecendo serviços de infraestrutura prontos e uma experiência de desenvolvimento visual e declarativa.

ADF Model: permite uma abordagem unificada para vincular qualquer interface de usuário a qualquer serviço de negócio, sem a necessidade de escrever código. É o principal módulo do framework.

Ele oferece:

- ADF Business Components: simplifica a construção de serviços de negócio;

- ADF Faces rich client: possui uma grande variedade de componentes de interface AJAX para aplicações JSF;
- ADF Controller: integra o JSF com o ADF Model. É uma extensão do JSF Controller com várias funcionalidades adicionais, como as Task Flows.

3 Componentes do projeto JSF

3.1 RichFaces

O projeto RichFaces é uma estrutura avançada de componentes de interface do usuário para integrar facilmente os recursos do Ajax em aplicativos de negócios usando o JSF, sua última atualização o RichFaces 4 baseia-se no suporte pioneiro do Ajax que começou com o RichFaces 3 e está padronizado no JSF 2.

Além de ampliar esses recursos do ajax, o RichFaces também melhora outras áreas do JSF 2, incluindo usabilidade, ajuste de desempenho, recursos dinâmicos, esfoliante e componente desenvolvimento. Isso permite aos usuários tirar o máximo proveito de todos os aprimoramentos de produtividade do JSF 2.

Disponibilizando:

- Um conjunto completo de componentes habilitados para o AJAX em duas bibliotecas a a4j que tem seus controles centralizados da AJAX e a Rico que componentes autônomos e prontos para usar;
- Validação do lado do cliente, expandindo a validação de feixe JSR 303 até o navegador;
- Encaminhamento avançado para combinar os requisitos de alto desempenho das aplicações empresariais do mundo real;
- Atualizações de componentes de envio, incluindo integrações do Serviço Java-Messaging (JMS) e vários mecanismos de transporte baseados no suporte do navegador;
- Possui o próprio kit de desenvolvimento de componentes (CDK);
- Documentação abrangente que aloca as melhores práticas de desenvolvimento e os detalhes dos componentes;
- Detalhadas e automatizados testes instalações de componentes, ações, ouvintes e páginas;
- Suporte abrangente para o navegador cruzado;
- Comunidade grande e ativa em sua fundação.

3.2 PrimeFaces

A Prime Technology não é um fornecedor de software, mas uma casa de desenvolvimento de software junto com as atividades de consultoria e treinamento. Uma estrutura que nem sequer é usada por seus próprios criadores pode facilmente perder pontos vitais em termos de usabilidade e simplicidade, uma diferença importante em comparação com os produtos do fornecedor é que em todos os projetos de seus

clientes o PrimeFaces é usado como o framework de front-end. Isso ajuda a visualizar o projeto do ponto de vista do desenvolvedor de aplicativos para que possa facilmente perceber os recursos perdidos e corrigir rapidamente os erros diferindo significativamente o PrimeFaces de outras bibliotecas.

3.3 OpenFaces

O OpenFace é uma implementação Python e Torch do reconhecimento facial com redes neurais profundas e é baseado no documento CVPR 2015 FaceNet, consiste em uma Incorporação Unificada para Reconhecimento de Rosto e Clustering por Florian Schroff, Dmitry Kalenichenko e James Philbin no Google. A tocha permite que a rede seja executada em uma CPU ou com CUDA.

Concedendo:

- Detecção rostos com modelos pré-treinados de dlib ou OpenCV;
- Transforma o rosto em códigos mandando para a rede neural;
- Usando uma rede neural profunda para representar (ou incorporar) o rosto em uma hipersfera de unidade de 128 unidades. A incorporação é uma representação genérica para o rosto de qualquer pessoa. Ao contrário de outras representações do rosto, essa incorporação tem a propriedade agradável de que uma distância maior entre dois embutidos de rosto significa que os rostos provavelmente não são da mesma pessoa. Esta propriedade torna as tarefas de agrupamento, similaridade de detecção e classificação mais fáceis do que outras técnicas de reconhecimento facial, onde a distância euclidiana entre recursos não é significativa;
- Aplica várias técnicas de classificação ou agrupamento preferenciais para os recursos para completar sua tarefa de reconhecimento;

3.4 IceFaces

O ICEfaces é um framework de desenvolvimento de aplicativos Rich Internet (RIA) de código aberto para Java EE. O ICEfaces funciona em plataformas que vão desde desktops a telefones inteligentes e Apple para Android. Ele melhora a eficiência do desenvolvedor, enquanto reduz o tempo ao mercado e os custos operacionais. São recursos e recursos ricos que permitem aos desenvolvedores fazer mais dentro dos limites de uma infraestrutura herdada do que pode ser imaginada.

Ela oferece escopos com várias formas de se trabalhar em um projeto como:

3.4.1 Automatic Ajax

O ICEfaces inclui vários recursos inovadores, como envio único, renderização direta para DOM e atualizações parciais de página que resultam cumulativamente no Ajax automático, eliminando completamente a necessidade de os desenvolvedores transmitirem as atualizações de página usando as tags padrão JSF `<f: ajax>`;

3.4.2 Ajax Push

A colaboração em tempo real é suportada no ICEfaces através dos mecanismos Ajax Push , o que facilita o envio assíncrono de atualizações de página para navegadores de clientes sem a necessidade de uma apresentação iniciada pelo usuário. Isso significa que os usuários do aplicativo podem ser informados instantaneamente sempre que ocorrer uma alteração de estado no aplicativo.

3.4.3 Resource Management

O ICEfaces amplia o framework JSF para auxiliar no gerenciamento de recursos. ICEfaces inclui novas anotações para ajustar o comportamento dos beans View-scoped. Embora o escopo View seja uma adição bem-vinda para gerenciar o ciclo de vida dos beans, o comportamento do escopo View pode não ser intuitivo em certos cenários. Especificamente, as anotações permitem que você controle o comportamento dos feixes de escala de visualização em navegação repetida para a mesma exibição e durante a exibição de disposição;

3.4.4 Scoped Resource Registry

O ResourceRegistry permite que um aplicativo registre instâncias javax.faces.application.Resource em tempo de execução. Cada Recurso está registrado em um escopo especificado (Aplicativo, Sessão, Vista, Flash ou Janela) para que o recurso possa ser coletado por lixo quando o escopo expirar;

3.4.5 Streamlined Configuration

O JSF 2 simplifica a configuração alavancando as anotações e os ICEfaces estão em conformidade com esta estratégia, simplificando a configuração em várias áreas. Os servlets específicos de ICEfaces e muitos dos parâmetros de configuração no ICEfaces 1.x não são mais necessários com ICEfaces. Isso torna ainda mais rápido e fácil adicionar ICEfaces ao seu aplicativo JSF e reduz consideravelmente a possibilidade de perda de produtividade devido a pequenos erros de configuração.

3.4.6 Window Scope

O ICEfaces apresenta um novo escopo personalizado chamado Window Scope, que é projetado para preencher uma lacuna nos escopos existentes disponíveis para o JSF 2, como ele existe para a vida de uma janela ou guia do navegador, incluindo sobreviver recarrega e atualiza.

3.5 EasyFaces

<http://www.easyfaces.com.br>

3.6 Gmaps4jsf

Ele visa integrar mapas do Google com o JSF. Os usuários do JSF também serão capazes de construir Panoramas e mapas complexos do Streetview com apenas algumas linhas de código (tags JSF). GMaps4JSF é uma das bibliotecas JSF Mashups que permite que os usuários do JSF criem aplicativos de mashup da Web 2.0 no JSF facilmente.

Ele permite manipular mapas como criações de mapa usando (latitude e longitude) ou (endereço), adicionar marcadores, adicionar textos de informações, adicionar controles, desenhar polígonos, polilinhas e círculos, eventos nos objetos, adicionar groundOverlay, executar operações diferentes, alterar entre tipos de mapas e etc.

4 Extensões JSF

4.1 PrettyFaces

O PrettyFaces é uma biblioteca open source de reescrita de URLs que facilita criação de URLs amigáveis (Friendly URLs), tornando suas páginas mais acessíveis para os usuários, possuindo um mecanismo que facilita a chamada de métodos através da URL no JSF.

4.2 OmniFaces

É uma biblioteca de utilitários que busca facilitar o desenvolvimento JSF para aplicações corporativas. Foi criada por Bauke Scholtz (BalusC) e Arjan Tijms, colaboradores regulares no popular site Stack Overflow de perguntas e respostas. Ele complementa as bibliotecas JSF existentes com várias classes auxiliares e soluções para problemas específicos. A amostra de componentes no site de demonstração do OmniFaces (live showcase), por exemplo, é baseada no PrimeFaces.

Ele cede:

- Destaca os campos que falharam na validação;
- Importação de valores de constantes para dentro do escopo da EL (Expression Language);
- Conversão automática de objetos do modelo em drop-downs e outros componentes de seleção;
- Valida todos os campos para valores como “todos ou nenhum”, “todos iguais”, “um ou mais” e “em ordem”, além de validações de “valor único”;
- RenderKit HTML5, que adiciona suporte a diversos atributos específicos do HTML5 para componentes UIForm e UIInput;
- Manipuladores de exceção Ajax: uma página de erro será exibida quando uma exceção ocorrer durante uma solicitação Ajax do JSF;
- Árvore com código de marcação personalizado por nível (utilizável para vários casos de uso recursivos);
- Coleções de funções da EL para lidar com arrays, conversões, datas e strings;
- Filtro de compressão GZIP para respostas HTTP;
- Funcionalidades para inclusão de Servlets e páginas JSP no Facelets.

5 Referências Bibliográfica

<https://imasters.com.br/linguagens/java/listagem-de-frameworks-java/?trace=1519021197&source=single>