

Sobre Java Server Faces

Flavio Pereira Marques

Curso Tecnológico em Análise e Desenvolvimento de Sistemas – FAQI – Unidade de Gravataí

51 3042-0000 – Gravataí – RS – Brasil

flaviomkes@gmail.com

***Resumo.** O artigo “Sobre JavaServer Faces” falará sobre a história desta ferramenta, como surgiu, evolução, popularidade, onde pode-se usá-la, suas vantagens e desvantagens, importância de seu conhecimento e utilização. O uso desta ferramenta vem crescendo e ajudando não somente empresas, mas também pessoas que trabalham como freelance na área de desenvolvimento de software. Sendo assim o leitor deste artigo terá uma breve noção de utilização da especificação, JavaServer Faces.*

1. Introdução

JavaServer Faces é um framework Java para desenvolvimento de aplicativos web focando na construção da interface do usuário, parte visual da aplicação que o usuário tem interação direta com o sistema ou máquina. Esta ferramenta tem por objetivo auxiliar o programador ou uma equipe de desenvolvimento de software, agilizando a construção do sistema. Isso possibilita que quem está desenvolvendo a aplicação ou o software tenha por objetivo concentrar-se na lógica em si do software.

Será visto em detalhes desde seu surgimento e evolução até os dias atuais, A especificação JavaServer Faces ou JSF como é mais conhecida tem suas versões, características e ciclos de vida, possui muitos recursos visuais para auxiliar no desenvolvimento de aplicações complexas.

2. História do JavaServer Faces

Esta especificação de desenvolvimento de interfaces foi lançada oficialmente em março de 2004 na sua versão ‘JSF 1.0’ e daquele ano até abril de 2013, versão mais recente, foram lançadas cinco versões desde 2004. Mas foi na versão 1.1 que o projeto emplacou devido a implementação da especificação do Java Community Process destacando maturidade e segurança. Com cada versão que estava sendo lançada, a ferramenta foi evoluindo e adquirindo melhorias em suas versões como: Correção de bugs sem mudanças na especificação, melhoramento nos sistemas centrais e da API (coincide com Java EE5), funcionalidade e desempenho melhorados facilitando muito sua utilização (coincide com Java EE6), pequenas mudanças na especificação para corrigir a versão 2.0 e introdução de novos conceitos. Aqui abaixo podemos ver melhor a data e a que cada versão evoluiu conforme seu lançamento.

- JSF 1.0 – 11/03/2004 => Lançamento da versão inicial;

- JSF 1.1 – 27/05/2004 => Correção de bugs sem mudanças na especificação;
- JSF 1.2 – 11/05/2006 => Melhorias nos sistemas centrais e na API (Application Programm Interface), coincidindo com Java EE5. Adoção inicial do Java EE;
- JSF 2.0 – 28/06/2009 => Melhorias para facilitar o uso melhorando sua performance;
- JSF 2.1 – 22/10/2010 => Correção de bugs da versão 2.0, aqui foi foram feitas pequenas mudanças nas especificações do framework;
- JSF 2.2 – 16/04/2013 => Introdução de novos conceitos;

3. O que é JSF (JavaServer Faces)?

JavaServer Faces é uma tecnologia nova se comparada com Java pois já tem seus 12 anos de existência desde seu lançamento oficial em 2004, seu foco é a construção de aplicativos web utilizando linguagem de programação JAVA, cuja sua especificação foi desenvolvida pelo grupo de peritos JSR-127 sob a supervisão do Java Community Process. Essa tecnologia é baseada em Servlet e JavaServer Pages(JSP). Um aplicativo JSF se parece com qualquer outro aplicativo servlet/JSP. Ele tem um descritor de distribuição, páginas JSP, bibliotecas de tags personalizadas e recursos estáticos. A interface de usuário de um aplicativo JSF é uma ou mais páginas que hospedam componentes JSF, tais como botões, caixas de entradas e formulários. Esses componentes são representados por tags personalizadas JSF e podem conter dados.

Para um programador escrever aplicativos JavaServer Faces ele deve ter conhecimentos nas tecnologias JavaServer Pages, servlet e também estar familiarizado com a tecnologia JavaBeans relacionadas e tags personalizadas. O JavaServer Faces é fortemente baseado nos padrões MVC e Front Controller. MVC é um, padrão de arquitetura que tem com objetivo isolar a lógica de negócio da lógica de apresentação e no Front Controller, todas as requisições do usuário são recebidas pelo mesmo componente, dessa forma, tarefas que devem ser realizadas em todas as requisições podem ser implementadas por esse componente.

4. Um pouquinho sobre Servlets e JSP (JavaServer Pages)

- Servlets – São apenas classes, claro escritas em java, que rodam em um servidor Web especial chamado contêiner Servlet/JSP ou contêiner Web, cujo seu objetivo é receber chamadas HTTP, processá-las e devolver uma resposta ao cliente.
- JavaServer Pages – Criada pela Sun, ela é uma linguagem de script que tem como objetivo gerar conteúdo dinâmico para páginas da internet. De acordo com a Sun, “A tecnologia JSP é uma extensão da tecnologia servlet criada para suportar assinatura de páginas HTML e XML”

5. Como JSF funciona?

Um aplicativo JSF funciona através de processamento de eventos disparados por componentes JSF sobre a páginas. Estes eventos são causados por ações do usuário. Por exemplo, quando o usuário clica em um botão, o botão dispara um evento. Ao disparar um evento, a notificação de evento é enviada via HTTP para o servidor. No servidor está

um servlet especial chamado FacesServlet, cada aplicativo JSF no contêiner web tem seu próprio FacesServlet.

Para as requisições JSF serem processadas, elas devem estar direcionadas para um servlet chamado FacesServlet. O FacesServlet cria um objeto chamado FacesContext, que contém a informação necessária para o processamento da requisição. O FacesContext contém os objetos: ServletContext, ServletRequest, ServletResponse que são passados para o método FacesServlet pelo contêiner web, neste processo o FacesContext é o objeto que é modificado.

6. Como criar um aplicativo JavaServer Faces:

A construção de um aplicativo JSF exige três etapas:

- Autografa páginas JSP, usando tags personalizadas que representam componentes JSF que serão renderizados como elementos HTML.
- Escrever JavaBeans como portador de estado da entrada do usuário e dos dados dos componentes.
- Escrever um ouvidor de eventos que determine o que deveria acontecer quando um evento ocorre, quando o usuário clica em um botão ou outro tipo de evento. O JSF suporta dois eventos:
 - ActionEvent – é acionado quando o usuário submete um formulário ou clica em um botão.
 - ValueChangeEvent – é disparado quando um valor em um componente JSF se altera.

A estrutura básica de uma página JSF é muito semelhante à estrutura de uma página HTML.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
" http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd ">
```

```
<html xmlns="http://www.w3.org/1999/xhtml"
xmlns:ui="http://java.sun.com/jsf/facelets"
xmlns:h="http://java.sun.com/jsf/html"
xmlns:f="http://java.sun.com/jsf/core">
```

```
<h:head >
<title>Titulo </title >
</h:head >
```

```
<h:body >
<h:outputText value="Estrutura básica de uma tela JSF" />
</h:body >
</html >
```

O conteúdo da página é definido no corpo da tag <html>. Esse conteúdo é dividido em duas partes: o cabeçalho, delimitado pela tag <h:head> (e não pela tag <head>), e o corpo, delimitado pela tag <h:body> (e não pela tag <body>).

As bibliotecas de tags que serão utilizadas para construir a página devem ser “importadas” através do pseudo-atributo XMLs aplicado à tag <html>. Observe no exemplo acima que as três principais bibliotecas do JSF foram importadas.

O grande poder do JavaServer Faces está no seu modelo de componentes de interface com o usuário, onde grandes aplicações são desenvolvidas com um conjunto de componentes que podem ser renderizados em diversos tipos de cliente, web, celular, etc.

Os principais componentes que a implementação de referência do JSF fornece são: formulários, campos de entrada de texto, campo de entrada de senha, rótulos com textos, textos de uma única linha, links, botões, mensagens, painéis, tabela HTML, tabela de dados (grid), coluna de tabela de dados, etc.

Atualmente existem diversas empresas que trabalham na criação de componentes personalizados. Como por exemplo, podemos citar a Oracle, com o ADF Faces Rich Client, IceSoft, com o IceFaces, JBoss (Redhat), com o Richfaces, e etc. Muitos desses componentes incluem múltiplos renderizadores para diversos tipos de clientes, tabelas avançadas, componentes para captura de datas e cores, menus suspensos, botões, barras de progressão, telas suspensas para seleção de valores e etc.

7. Vantagens e Desvantagens dessa Tecnologia

7.1. Vantagens

O JavaSever Faces torna o processo de desenvolvimento mais rápido, essa tecnologia fornece recursos que solucionam problemas práticos em desenvolvimento Web:

- Componentes fáceis de usar e interface de usuário extensível
- Fácil navegação de página
- Validadores de entrada
- Gerenciamento de JavaBeans
- Manipulação de evento
- Fácil manipulação de erro
- Suporte para internacionalização

7.2. Desvantagens

Mas como tudo tem suas vantagens e desvantagens agora apresentaremos as desvantagens desta tecnologia que por sinal tem um número considerável. E dependendo do tamanho e complexidade do projeto são pontos a considerar a sua utilização.

- Curva de aprendizagem, projetos pequenos programadores sem conhecimento da ferramenta e prazos curtos
- Documentação, inferior ao JavaServer Pages(JSP)
- Opacidade na execução, debug, otimizações
- Suporte ao desenvolvimento comparado com JavaServer Pages
- Rigidez da abordagem, Model View Controller (MVC)

Conclusão

Neste artigo foi feito um breve relato sobre JavaServer Faces (JSF), que nada mais é do que um framework focado na interface do usuário, ou seja, a parte bonita do sistema.

Como foi mencionado, esta tecnologia foca na interface do usuário, do sistema fazendo com que o programador concentre-se na lógica do sistema. Aqui pôde-se ver a importância desta tecnologia, mas também foi visto que ela tem suas desvantagens, e se for muito utilizada por um programador que está iniciando sua carreira, pode gerar certa comodidade na pessoa fazendo com que ela “desaprenda” a programar. Mas mesmo assim esta tecnologia está cada vez mais popular e utilizada por desenvolvedores e empresas. Ela não é indispensável, sua utilização dependerá do projeto a ser desenvolvido e o nível de conhecimento de quem irá usar.

Referências

Jorge Willian Rodrigues. “Introdução ao JSF”. Disponível em: <http://pt.slideshare.net/JorgeWilliamRodrigues/introducao-a-jsf> . Acesso: 09/2016

Alessandro Jatoba. “Primeiros passos com JavaServer Faces”. Disponível em: <http://www.devmedia.com.br/primeiros-passos-com-javascript-faces/33272>, Acesso: janeiro/ 2016

Kurniawan, B. (2004) Programando em JavaServer Faces. Rio de Janeiro: Editora Ciência Moderna Ltda, 492 p.

Mojarra. “JavaServerFaces”. Disponível em: https://pt.wikipedia.org/wiki/JavaServer_Faces, maio /2016.

CAELUM. “Apostila do curso FJ-22 - Introdução ao JSF e Primefaces”. <http://www.devmedia.com.br/introducao-ao-java-server-pages-jsp/25602> . Acesso: setembro/ 2016

Thiago Faria. “Java EE7 Com JSF, Prime Faces e CDI”; 2ª Edição. Agosto/ 2016.