JSF TAG LIBRARY

Tyler O’Riley

CSD430 Assignment 10

12/10/2023

JSF or Java Server Faces is a form of Java like JSP. The primary difference between the two is how JSF utilizes tag libraries. With JSP, there is a need for HTML script and CSS code when constructing the visual display for users. JSP is already a shortcut in modern code by allowing a format where HTML and Java servlets can be written side-by-side when being run on a server. JSF is an even newer form aiming to use built in taglibs that render just like HTML to produce the webpage for clients to use. At first this topic may sound a bit confusing so we will cover how these two file types differ.

As we mentioned above, JSF files no longer need HTML tags as this file type already has a tag library it can utilize by importing the library (found in links provided). Though JSF has these built in tags, it is not a replacement for HTML, rather a complement with the aim of making coding easier. Like software development over the past 30 years, we are always striving to produce new ways to code that take less time and less script. JSF accomplishes this by having an XHTML file that runs cohesively alongside a java bean file. In normal format, a java bean file would need to be run alongside HTTPS requests to initiate the needed business requests between the JSP and HTML. With JSF files, the java bean file can conduct its query with the XHTML file as the JSF code allows for these requests to be conducted within the file, bypassing the need for HTTPS requests. This change alone allows for paragraphs of code to be saved by eliminating the need for HTTPS test and retrieval codes. In this context you can see where some code can be freed up, but not necessarily replaced. A good JSP file still works well in context to JSF, it is just good to consider where JSF may create ease of access where the extra HTML may cause larger script/load times.

First, let’s look at the JSF Core library to explore how simple HTML can be made easier when written in this context. It is important to note as we cover these basics where and when we leave out HTTPS script. As stated, with the use of JSF we bypass this need and call functions directly. When having a user create a username, we would first use the h:inputText tag to create our line for the user input. From there, the main difference will be the reference for the java bean request. We would simply use syntax as {user.name} and an action of “response.xhtml” to call the user bean file to submit the users name onto the response page by using direct requests from the managed bean file, the index file, and the response file. The feature code for this can be found in the attached files. I opted to use our reading example as it was easy to follow and understand.

Finally, we will review The Facelets library as well. Without going into detailed code, we will review the library based on its function. JSF is a collection of tags primarily to help with HTML functions such as button and text requests. Facelets are an extension of JSF to assist with the view capabilities of JSF. The tag library assists with XML code to properly compose and display each view as needed. Today Facelets are usually included with JSF libraries but before that it was a needed extension. When Facelets did not exist, JSP relied on the handle view control for JSF which proved troublesome with compatibility issues. Facelets were specifically designed to eliminate this issue with its designed library to assist the JSF library.

RESOURCES

*Standard JSF Tags | An overview of the JSF Core Tags | InformIT*. (n.d.). https://www.informit.com/articles/article.aspx?p=1606899

*JSF - template tags*. (n.d.). https://www.tutorialspoint.com/jsf/jsf\_templates\_tag.htm

*What is Facelets? - The Java EE 6 tutorial*. (2013, January 1). https://docs.oracle.com/javaee/6/tutorial/doc/gijtu.html

Darshan University. (2020, April 29). *Unit 5.3: JSF Basic Tags* [Video]. YouTube. https://www.youtube.com/watch?v=H\_Ji0L3seis

*JSF Example - javatpoint*. (n.d.). www.javatpoint.com. https://www.javatpoint.com/jsf-example