

Course code: **JSF2**

Course title: **Advanced mechanisms of user interface development in the Java Server Faces technology**

Days: 3

Description:

Course intended for:

The training is intended for programmers having practical experience with Java Server Faces, wishing to get familiar with more advanced techniques of working with JSF.

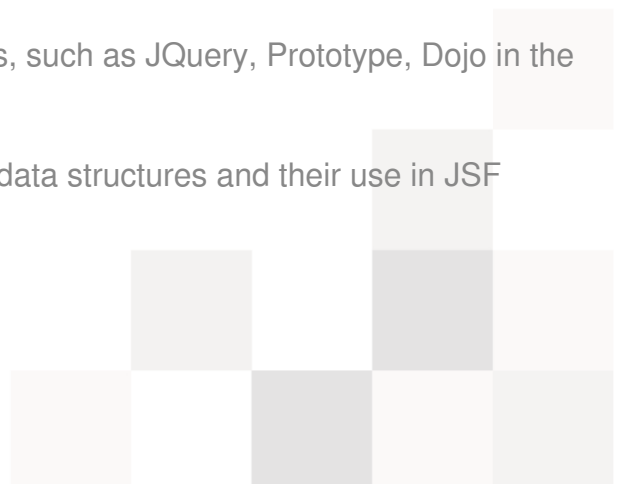
Course objective:

The training objective is to get familiar in practice with the advanced programming techniques, allowing for solving of the most complex problems encountered during development of JSF based applications.

Key issues:

- Code development using the DRY rule (in particular, own components and their re-use)
- Development of own validators, converters and events
- Handling of data validation and display in the parent-child correlation (having many choice lists in a single form, which are correlated)
- Navigation, in particular, development of bookmarks (bookmarkable URLs/buttons) and URLs friendly for users and browsers (user-friendly URLs, SEO-friendly URLs)
- Use of JavaScript (JavaScript, JQuery, ClientBehavior) for development of a more complex user interface
- A practical comparison of Javascript libraries, such as JQuery, Prototype, Dojo in the context of their use in a JSF application
- Development of JavaScript-friendly (JSON) data structures and their use in JSF

Course strengths:



The training is focused on issues, which are very often perceived as being problematic, difficult to program using JSF. Every programmer, who has encountered JSF, sooner or later finds out that JSF, like every other technology, has its strengths and weaknesses, and not everything can be done automatically, using the basic JSF mechanisms. The ability to solve certain groups of problems requires some in-depth knowledge on the functioning of JSF, and this training is aimed at providing the necessary information in this regard.

Requirements:

The training participants are required to have the Java language programming skills (to be learned at the course J/JP), to be familiar with basic issues of Java Server Faces (to be learned at the course J/JSF2) and to have basic knowledge of Javascript.

Course parameters:

4*8 hours (4*7 net hours) of lectures and workshops (with a visible emphasis on workshops).

Group size: no more than 8-10 participants

Course curriculum:

1. Introduction

- A summary of the basic JSF concepts
- Practical encounter with the application, which is to be a starting point for the exercises completed during the training
- A summary (using the application, whenever technically possible) of the key issues and concepts associated with JSF and JEE
 - The role of JSF in the world of JEE, alternative solutions (JSF vs. Struts2, JSF vs. GWT, JSF vs. JSP)
 - Available implementations (RichFaces, PrimeFaces, IceFaces etc.)
 - MVC, MVP, CDI
 - JSF component model
 - Life cycle
 - ManagedBeans



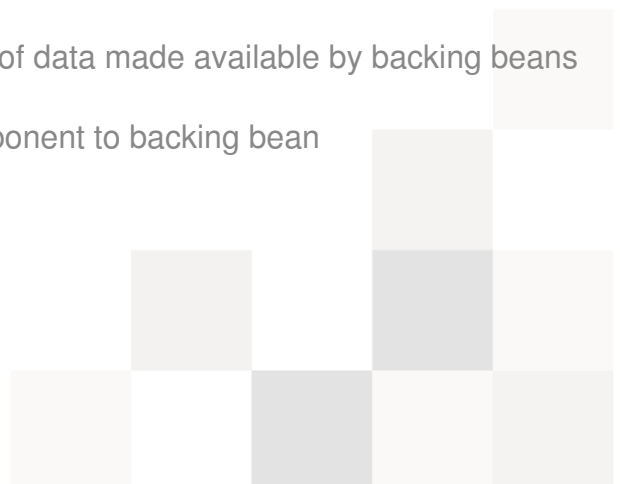
- Navigation
- Validation
- Facelets
- Introduction to key issues discussed during the training
 - Composite Components
 - JavaScript (jQuery, Ajax, JSON, Client Behaviors)
 - Navigation

2. Development of components - basic information

- Practical introduction to key concepts necessary to develop own components
 - A component or a renderer?
 - Composite namespace
 - Tags: composite:interface, composite:attribute, composite:implementation
- Development of simple components using knowledge acquired so far – one component, many opportunities to use it
 - Components using input type controls (data input)
 - Components using select type controls (selection of data from choice lists)
- RichFaces CDK

3. Development of components – advanced issues

- Component cooperation with logic on the server side
 - Obtaining by the component of data made available by backing beans
 - Data transmission from component to backing bean
- Development of own events
 - Event queuing



- Initiation events on the server side (Action listeners)
- Events associated with effects of events initiated at the user interface level (Event listeners)
- Events associated with JSF life cycle phases - Phase Listeners
- Own events and JSF life cycle phases
- Validation and conversion
 - Development of own validators and converters
 - Overriding of full validations ("Cancel" type activities, the immediate attribute and its impact on request processing, conditional, partial validation)
 - BeanValidation (JSR303) and JSF
 - The standard life cycle of JSF and the necessity to modify it in practice
- Parent-child correlations in the world of JSF – dependencies between lists of choice (availability/unavailability depending on selection of value in the parent/child list)
- Intelligent data acquisition (e.g. pagination in tables or complex structures such as trees)

4. Use of Javascript, Ajax

- Broadening of the component capabilities by using Javascript
 - Development of JavaScript friendly structures (JSON)
 - JSON in JSF applications
- Ajax
 - Data acquisition techniques on the Web side (Pull, Comet aka Push/Reverse Ajax)
 - Controlling of the order of transmission of requests (Callbacks, Queuing)
 - Rendering of page fragments (Partial Rendering, Partial Requests)

- Use of mechanisms contained in available implementations of the JSF standard (RichFaces, PrimeFaces)
 - „a4j" library in RichFaces
 - Implementation of „Ajax-like" behaviors using PrimeFaces
 - Use of JQuery – the advantages of this solution
 - Enriching of user interface through introduction of "special effects" (e.g. Fading, Sliding)
5. Development of a JavaScript code that can be re-used
- Use of ClientBehavior and ClientBehaviorHolder for implementation of a reusable JavaScript code
6. Navigation (with use of PrettyFaces)
- Bookmarks (bookmarkable URLs/buttons)
 - User and SEO friendly URLs
 - View Parameters
 - Back button problem
7. Communication in JSF based on events
- Traditional approach to communication between objects
 - The Observer pattern in implementation of communication mechanisms
 - Use of CDI for communication between objects (implementation of an „EventBus" type solution)
8. Summary

