

JAV250 : Java EE Web Development met JSF

Code : JAV250

Duur : 4 dagen

Categorie : Java EE

Doelgroep :

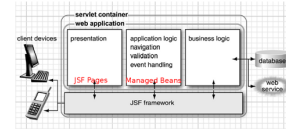
Developers die Java Web applicaties met het standaard JSF 2.x Framework willen bouwen.

Voorkennis :

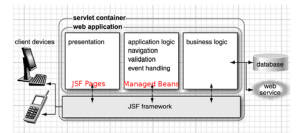
Om aan deze cursus te kunnen deelnemen is ervaring met programmeren in Java en object oriëntatie vereist. Voorkennis van Java EE Web Applications is bevordelijk voor een goede begripsvorming.

Uitvoering :

De theorie wordt behandeld aan de hand van presentatie slides en wordt afgewisseld met oefeningen. Demos zorgen voor een verheldering van de theorie.



Java EE Web Development with JSF 2.0



Inhoud :

Deze cursus behandelt hoe moderne state of the art Java EE web applicaties worden gebouwd met het Java Server Faces (JSF 2.x) framework. De cursus begint met de fundamenteën van de JSF Framework dat is gebouwd bovenop het Java Web Programming model en gebruik maakt van het MVC pattern met een Front Controller als centraal element. De architectuur van de JSF Framework, dat een statefull component model op de server gebruikt, wordt besproken. Vervolgens wordt aandacht besteed aan de rol van het Faces Servlet en de verschillende fasen van de JSF Request Life Cycle. De rol van het centrale configuratie bestand faces-config.xml wordt uitgelegd en ook worden de navigation rules in JSF en de rol van de managed beans behandeld. Vervolgens wordt de structuur van een JSF pagina met Facelets, het gebruik van de Expression Language en het gebruik van de verschillende UI componenten en custom tag libraries besproken. Event handling uit de UI componenten krijgt de aandacht en ook de validatie en conversie van input uit de User Interface met behulp van JSF built-in of custom validators en converters passeert de revue. Daarna wordt er besproken hoe het moderne asynchrone Ajax request model kan samenwerken met JSF. Tot slot wordt aandacht besteed aan het creëren van custom components, facelet templates en het gebruik van view parameters.

Module 1 : JSF Intro

- JSF Background
- Servlet and JSP Problems
- Classic MVC Pattern
- Model 2 Architecture
- Struts Framework
- JSF Framework
- JSF Feature Overview
- Server Side UI Components
- JSF Application Architecture
- Request Processing Phases
- JSF Implementation
- JSF API
- JSF Libraries on Google Trends
- Roles in JSF

Module 2 : JSF Basics

- Choose JSF Implementation
- Deployment Descriptor
- PROJECT_STAGE Development
- Faces Configuration File
- Facelets Page Structure
- Managed Beans
- @ManagedBean Annotation
- Hello JSF
- Default Navigation
- Facelet Default Navigation
- Handling Request Parameters
- Analyzing Request Parameters
- Facelet Input Form
- Selecting Result Pages
- Common JSF Problems
- JSF Tradeoffs

Module 3 : Managed Beans

- Java Beans
- Bean properties
- Accessor Usage
- Encapsulation Benefits
- Managed Beans
- Parts of Managed Beans
- Placeholder for Results
- Dual Role Bean Properties
- Textfields and Checkboxes
- Prepopulating Input Fields
- Naming Beans
- Bean Scopes
- Bean Scope Annotations
- Application and Session Scope
- Raw Request and Response
- Dependency Injection
- ManagedProperty

Module 4 : Navigation

- JSF Navigation
- Static Navigation
- Navigation Rule Syntax
- Explicit Mapping Considerations
- Dynamic Navigation
- Dynamic Navigation Logic
- Explicit Dynamic Navigation Rule
- Bean Declarations
- Bean Declaration Considerations
- Configuring Managed Beans
- Sharing Result Pages
- Using Wildcards
- Precedence for Wildcards
- Omitting from-outcome
- Conditional Navigation Rules
- Computed to-id's
- Common Navigation Problems
- Common Errors

Module 5 : Properties

- Properties
- Using Property Files
- Using Subfolders
- Facelet using Properties
- Parameterized Messages
- Parameterizing Display
- Using f:param
- Using Parameterized Properties
- Internationalized Messages
- Localizing Strings
- Using Localized Messages

Module 6 : Expression Language

- Expression Language
- Advantages EL
- JSF versus JSP EL
- Outputting Bean Properties
- Nested Bean Properties
- Submitting Bean Properties
- Getter versus Setter Calls
- Accessing Collections
- Using Square Brackets
- Implicit Objects
- EL Operators
- Conditional Text
- Rendered Attribute
- Method Arguments

Module 7 : Event Handling

- User Interface Events
- Event Handlers versus Ajax
- Types of Event Listeners
- ActionListener in Facelets
- Implementing ActionListener
- Change CSS Style
- Change Style Input Form
- Change Style Bean
- Bean for Action Controller
- Setting the Locale
- Locale from User Settings
- Switch Locale
- ValueChangedListener

Module 8 : Validation

- Need for Validation
- Validation Approaches
- Manual Validation
- Standard Error System
- Creating Error Messages
- Displaying Error Messages
- Manual Validation in Bean
- Action Controller Method
- Alternative Approach
- Implicit Automatic Validation
- Precedence Validity Tests
- Explicit Automatic Validation
- Format Validator Tags
- Conversion versus Validation
- Custom Validator Methods

Module 9 : Ajax

- Ajax Web Applications
- Ajax in JSF
- Simple JSF Ajax
- General Form f:ajax
- Ajax Facelets Pages
- Ajax Controller Methods
- render Attribute
- Facelet Non-Ajax Version
- Facelet Ajax Version
- execute Attribute
- Facelets Code execute
- event Attribute
- Default Events
- onevent Attribute
- Limitations h:outputText

Module 10 : Data Tables

Handling Variable-Length Data
h:dataTable
Using h:dataTable
h:Column
Headers and Captions
StyleSheets
Ajax-enabled tables
Tables conditional values
Looping with ui:repeat
Conditional Text

Module 13 : View Params

View Params
Why View Params
Using View Params
Typical Page Structure
Capture Request Parameters
h:link and h:button
Outgoing Request Parameters
Using non-JSF forms
JSF page use f:viewParam
Using POST-redirect-GET
Bookmarkable Result Pages

Module 11 : Composite Components

Composite Components
Component Definition
Using Components
Components with Attributes
Using Composite Attribute
Component Structure
Using ui:repeat
Relative URL's
Nested Components
Component Files

Module 12 : Facelet Templating

Page Templating
Using Templates
Namespaces Template Files
Template File Details
Template File Example
Client File Details
Client File Example
Including Files
Using ui:include
Snippet Files
Handling Relative URL's