

## **Java or .NET Plumtree Enterprise Portlet Development and UI Customization**

### **Course Summary**

#### **Description**

This course is an all-inclusive portlet development course that will cover basic through advanced Plumtree Portlet development concepts. Although there is a small lecture component, the course is primarily exercise-based. Students who meet the prerequisites and who successfully complete all of the exercises will leave the course ready to design and write enterprise class Plumtree Portlets.

#### **Topics**

- Introduction to Plumtree, the MPPE, Portlets & Java
- Introduction to Portlet Frameworks and Database-driven Portlet
- The Plumtree Remote Client
- Database Schema
- Intercepting Events and Executing Custom Code

#### **Prerequisites**

Students should have a basic understanding of the Plumtree Portal, MPPE & Portlets. They should also be proficient in Java and Web Programming. Experience with Eclipse helpful

#### **Duration**

Five days

## **Java or .NET Plumtree Enterprise Portlet Development and UI Customization**

### **Course Outline**

#### **I. Introduction to Plumtree, the MPPE, Portlets & Java**

- A. Demonstration of Plumtree Server and Integration Products
- B. Install a Java IDE and come up with an idea for a new Portlet
- C. Design a Plumtree Portlet and create a GSO and Gadget Object; add to MyPage and Community HTTPGP & the Gateway
- D. Yahoo! Search Portlet: Design a portlet that allows the user to issue a search in Yahoo!
- E. TCP Tracing: Use a free utility to trace TCP activity on port 80 (HTTP); observe CSP in action
- F. Gadget Development Kits
- G. Gateway Specific Configuration
- H. Design a Plumtree Portlet that uses a Gateway Setting to display the phrase Hello <current user name>!
- I. Introduction to Settings
- J. Change Font Color: Use a Gadget Setting to allow the user to personalize the Portlet with a font color
- K. My Bookmarks: Design a Plumtree Portlet that displays an end-user customizable list of bookmarks to web sites
- L. Settings Review
- M. Community Bookmarks: Design a Plumtree Portlet that allows the community manager to set a customizable list of bookmarks to web sites

#### **II. Introduction to Portlet Frameworks and Database-driven Portlet**

- A. Design
- B. Database Schema: Design database schema for a Data Entry and Browse Framework Portlet
- C. Administrative Settings Review
- D. Administrative Settings: Design the Administrative Preferences page for a Data Entry and Record Browse Framework Portlet
- E. Data View/Entry Form: Design Data View/Entry Form for a Data Entry and Record Browse Framework Portlet
- F. Record Browsing: Design Record Browsing for a Data Entry and Record Browse Framework Portlet
- G. Portlet Caching, Performance and Scale-ability
- H. Caching: Add ETAG/IF-NONE-MATCH caching to Data Entry and Record Browse Framework Portlet
- I. Enhance Record Browsing with Pagination, Sorting and Filtering
- J. Internationalization/Localization
- K. Create a Localized Portlet

#### **III. The Plumtree Remote Client**

- A. Using the PRC: Obtaining information from the server
- B. Markup Tags: Displaying links to administrative editors
- C. IFrames: costs and benefits
- D. Implementing In-place Refresh with IFrames

## **Java or .NET Plumtree Enterprise Portlet Development and UI Customization**

### **Course Outline (cont'd)**

#### **IV. Database Schema**

- A. Design database schema for a Data Entry and Browse Framework Portlet
- B. Administrative Settings Review
- C. Administrative Settings: Design the Administrative
- D. Preferences page for a Data Entry and Record Browse Framework Portlet 1
- E. Data View/Entry Form: Design Data View/Entry Form for a
- F. Data Entry and Record Browse Framework Portlet
- G. Record Browsing: Design Record Browsing
- H. Data Entry and Record Browse Framework Portlet
- I. Caching: Add ETAG/IF-NONE-MATCH caching to Data Entry and
- J. Enhance Record Browsing with Pagination, Sorting and Filtering
- K. Internationalization/Localization
- L. Create a Localized Portlet

#### **V. Intercepting Events and Executing Custom Code**

- A. Building a Class that Executes on User Login
- B. Structure of the 5.0 User Interface
- C. Compiling the UI
- D. Modifying the UI
- E. Modifying the Login Page