

Développer un client riche Java avec Swing

Le projet « Swing » ou Java Foundation Classes (JFC) ne date pas d'hier : il est né d'une initiative Netscape en 1997 pour doter Java d'un framework permettant de créer des IHMs riches. Swing est alors adapté à l'écriture de « clients lourds » car son architecture, les performances des PC et d'Internet permettent difficilement de l'utiliser comme un « client riche ». Pendant longtemps, le client lourd a été oublié, au profit de l'application web et de HTML avec les limitations d'ergonomie qu'on lui connaît à l'arrivée du « web 2.0 ». Disposer d'une interface riche, ergonomique et fonctionnelle est re-devenue une exigence des utilisateurs.

Bonne nouvelle, depuis 1997 les PCs sont devenus bien plus puissants, les réseaux à « haut débit » et les architectures matures. Dès lors, imaginer une interface client riche écrite entièrement en Java, téléchargeable via Java Web Start et dialoguant avec un serveur d'application Java EE est une bonne alternative à Ajax, JavaScript et autres techniques « 2.0 » ?

Programme

Présentation de Java Swing

- Le framework AWT (Abstract Window Toolkit)
- Historique de Swing
- Concepts architecturaux de Swing
- Organisation des packages et vue générale de Swing
- Architecture n-tiers, serveur d'applications Java EE et positionnements possibles d'un client lourd/riche Swing

Panorama et classification des composants

- Conteneurs
- Contrôles simples
- Gestionnaires de placement
- Composants graphiques complexes

Les conteneurs Swing

- La classe racine JComponent
- JFrame, JDialog, JApplet, JPanel, JScrollPane, JTabbedPane

Les composants graphiques de base

- JButton, JTextField, JTextArea, ?
- Images, Icones, Placements
- Propriétés communes et spécificités

Les gestionnaires de disposition (Layout Manager)

- Configuration de la position et des dimensions à l'intérieur d'un container
- Les principaux Layout : BorderLayout, FlowLayout, GridLayout, GridBagLayout
- Création d'un Layout spécifique

Le Look & Feel

- Choix du Look & Feel
- Créer son propre L&F

Le modèle de programmation événementielle

- Les différents types de Listeners et Events Swing
- Implémentation de Listener, héritage d'Adapter et inner classes

Composants graphiques avancés

- Les menus
- Les onglets (TabbedPane)
- Le conteneur MDI (fenêtres internes multiples)
- Les boîtes de dialogue
- Les composants (Navigateur HTML, Editeur RTF, Gestion des couleurs, Sélection de fichier)

Composants graphiques complexes

- Implémentation MVC
- Classes de vues avancées : JList, JTable, JTree (arbres)
- Classes de modèles prédéfinies (TableModel, ListModel, TreeModel)

Swing avancé

- Intégration des threads avec Swing
- Les techniques single-thread et multi-thread
- Déploiement d'applications avec Java WebStart

Détails

Code :

JR-JFC

Durée :

3 jours

Objectifs :

- Positionner Swing dans une architecture n-tiers Java
- Connaître l'étendue des possibilités d'IHM offertes par Swing
- Être capable de développer une application graphique avec Java Swing
- Maîtriser le MVC et le modèle de programmation événementielle de Swing
- Savoir déployer une application Java avec le Java Web Start

Public :

- Ingénieurs
- Architectes
- Chefs de projets

Pré-requis :

Très Bonne pratique de Java

Tarif : 1290 € H.T. par personne (déjeuner inclus)