

Réf : T475-020

Durée : 5 jours

Public

Développeurs, chefs de projets souhaitant créer une application internet riche (RIA).

Vous serez capable de

- Construire une application avec Flex Builder ;
- Utiliser les principaux composants d'interface ;
- Charger et manipuler des données XML.

Pré-requis

- Avoir suivi le cours « T475-010 – Programmation orientée objet avec le langage ActionScript 3.0 », ou avoir les connaissances équivalentes, ou maîtriser un langage de programmation objet (C++, Java, C#...) ;
- Connaître la syntaxe XML.

CONTENU PEDAGOGIQUE

Introduction à Flex 3.0

- Applications Internet Riches (RIA)
- Framework Flex SDK, Flash Player et Machine virtuelle AIR

Flex Builder

- Environnement de développement Flex Builder basé sur Eclipse
- Editeurs de code, vues et perspectives
- Création et configuration d'un projet
- Création d'une première application
- Lancement d'une application
- Conception en mode Design
- Exécution en mode Debug

Construction d'une application simple

- Utiliser les contrôles utilisateurs
- Intégrer des contrôles de texte
- Ajouter des éléments Button, LinkButton et PopUpButton
- Utiliser les autres contrôles CheckBox, Radio, NumericStepper, DateField, ColorPicker
- Etablir des liaisons de données (Binding) entre composants
- Afficher des images embarquées ou distantes

Gestion des événements

- Comprendre la notion d'événements
- Intercepter et traiter les événements
 - Phases de capture, cible et de remontée
 - Objet target et currentTarget
- Découvrir les classes Event, MouseEvent et FlexEvent

Mise en page de l'application

- Structurer l'application par imbrication de conteneurs
- Utiliser des conteneurs de positionnement verticaux (VBox) et horizontaux (HBox)
- Fixer les contrôles avec les conteneurs de positionnement absolu Canvas
 - L'attribut layout
 - Les contraintes de positionnement absolu
- Construire un module avec le conteneur Panel
 - Configurer le module
 - Ajouter une barre de contrôles
- Positionnement et dimensionnement des contrôles

• Création de composants personnalisés MXML

- Déclaration et instanciation des composants
- Déclaration d'une interface (attributs/méthodes) publique
- Création d'événements internes
- Diffusion des événements propres au composant

Animation de l'application

- Définir des états (State) en mode Design et en mode source
- Déclencher les changements d'états
- Associer des effets de transitions entre états

Mise en forme de l'application

- Découvrir les propriétés de style
- Définir et appliquer un style CSS à un composant
- Utiliser des skins
- Appliquer un thème

Travailler avec des données

- Utiliser les types primitifs Array et Object
- Manipuler des données XML et XMLList avec E4X
- Mise en place de fonctions de tri et de filtre avec les composants ArrayCollection et XMLListCollection
- Configurer les composants d'affichage de données (List, TileList, DataGrid, Tree)
 - Formatage des données avec le composant ItemRenderer
 - Edition des données avec le composant ItemEditor

Navigation dans l'application

- Utiliser le conteneur ApplicationControlBar
- Intégrer une barre de menu et définir un menu contextuel
- Utiliser les conteneurs de navigation prédéfinis Accordion et TabNavigator
- Personnaliser la navigation avec les composants ViewStack, LinkBar, ButtonBar et TabBar

Mise en œuvre de la validation de données

- Utiliser les composants Validator prédéfinis
- Configurer l'affichage des erreurs de validation
- Mise en place de validateurs personnalisés utilisant les expressions régulières
- Gestion de l'événement valid

Internationalisation de l'application

- Créer des fichiers de langue et configurer le projet
- Gérer de ressources localisées
- Mettre en place un contrôle dynamique de choix de langue

Chargement de données distantes

- Création et configuration d'un service HTTPService
- Gestion du résultat avec l'événement ResultEvent
- Gestion des erreurs avec l'événement FaultEvent
- Mise en place d'une liaison de données entre le service http et un composant