



Développement Swift iOS iPhone iPad

Développement d'Applications Swift iOS pour iPhone iPad

Durée : 5 jours / 35 heures

Mode : en inter ou intra-entreprise

Objectifs Pédagogiques / Description

Cette formation permet aux développeurs Java, C#, C++, AS3, PHP etc... d'acquiesir toutes les notions pour réaliser avec le framework Cocoa Touch et l'outil Xcode d'Apple des applications mobiles aussi bien pour les téléphones iPhone que pour les tablettes iPad.

Cette formation a pour objectif d'apprendre à développer des applications pour iPhone et iPad avec les outils d'Apple. Elle permettra de connaître le frameworks et le langage pour créer une application iOS. Le stagiaire apprendra à connecter les données de son entreprise à une application iPhone ou iPad et à utiliser les bonnes pratiques pour structurer son projet iOS.

Le module aborde les sujets principaux du développement iOS : les bases du langage Swift, l'organisation MVC de Cocoa Touch, les storyboards, la navigation, l'autolayout, les composants d'interface, les listes et leurs personnalisations, l'accès serveur REST/JSON, Core Data et SQLite, la personnalisation de l'App, la gestion des certificats, le déploiement sur iTunes Connect.

Le stagiaire réalisera tout au long de cette formation sa première application pour Mobile depuis la conception en prototype papier jusqu'à la réalisation sur un projet concret. Les thèmes de l'ergonomie, des patterns d'IHM et de l'organisation d'un projet multi-écrans sont ainsi concrètement mis en pratique.

Programme

Hello Cocoa Touch

- Technologies de développement iOS : langage, framework, outils, workflow
- Lignes de conduite pour les interfaces (IHM) mobiles
- Découverte des storyboards, Structure d'un projet Swift,
- L'organisation Modèle Vue Contrôleurs (MVC) d'Apple
- Les concepts outlet, target et actions
- Simulateur, certificats et déploiement sur un device

Fondamentaux Swift

- Playground ; aperçu du langage ; type de base ; type inféré ; espace de nommage
- Manipulation de String, print et la console ; formats d'affichage,
- Valeur optionnelle ; boucle, condition ; les fonctions
- Les collections : tableaux et dictionnaires

Un peu plus de Swift

- Classe, propriété, méthode, héritage ;
- Constructeur, getter, setter, observer, appel de méthodes et propriétés ;
- Gestion de la mémoire avec Swift ; les structures
- Foundation Kit et classes internes
- Aperçu d'Objective-C : interconnexion Swift-ObjC

Atelier Prise en main d'Xcode

- Création et organisation du projet, Gestion des éléments, storyboard
- La notion d'assistant, simulation des tailles d'écrans (résolution des écrans),
- création d'éléments (vues, contrôleurs, classes) ;
- Les différentes vues de l'outil de développement ; Propriétés du projet
- Git, création, commit
- Le debug ; les raccourcis, La documentation

Audience

Développeurs, Chefs de projet ayant des notions de programmation et souhaitant construire des applications iPhone et iPad

Pré-requis

- Notions de programmation ; typiquement Développeurs Java, C#, AS3, C++, C, Dephi, PHP, JavaScript...

- Disposer d'un ordinateur Mac équipé de la dernière version d'OSX

Programme (suite)

Storyboarding

- Storyboard, UIViewController & UIViews, scènes et enchaînements (segue) ;
- Type de segue, API de pilotage ; unwind ; rappels sur outlet, target et actions
- Composants de navigation : UINavigationController et UITabBarController
- Cycle de vie des contrôleurs (prepareForSegue, viewDidLoad, viewWillAppear...)
- UIView : positionnement

Autolayout

- Springs and struts dans interface Builder
- Autolayout : activation ; attributs et opérateurs de contraintes ;
- Appliquer l'autolayout dans les storyboard ;
- Fixer une contrainte en code avec NSLayoutConstraint ;
- Priorités, compression, conflits, animer une contrainte

Catalogue de composants

- Hiérarchie des composants : UIView, UIControl, bouton, label, switch, slider...
- Gestion du texte : UITextView, NSAttributedString, couleur avec UIColor
- UIImage et UIImageView ; modes, gestion des catalogues d'images, slicing, couleur-template
- UIDatePicker, gestion des dates
- Les protocoles Swift, utiliser un protocole, exemple avec la délégation

Liste, Collection et Cellule

- UITableView ; dataSource et delegate
- Création dans Xcode ; Customisation de la liste et des cellules ;
- Alimentation de la liste avec des données dynamiques ; sélection et navigation
- Les collection génériques avec UICollectionView ;
- Scroller avec UIScrollView

Récupérer des données serveur

- Créer des requêtes HTTP REST JSON
- Gérer le mode asynchrone sur iOS avec la notion de closure
- Créer des objets métier à partir de données JSON
- Comprendre le multithread avec la notion de queue et synchroniser des éléments d'interface sans bloquer le thread principal
- Utiliser une bibliothèque externe comme Alamofire

Structurer l'accès aux données locales

- Système de fichiers ; Lecture et écriture
- Organiser les données avec Core Data et SQLite ; comprendre la persistance objets
- Créer des DataModel et des classes métiers en utilisant les extensions Swift
- Contexte d'édition et sauvegarde automatique

Personnaliser son application

- Changer les teintes des éléments avec Tint
- Inclure des images de fond ; le protocole UICollectionAppearance
- Créer des animations simples
- La reconnaissance des gestes : UIGestureRecognizer

Storyboarding avancé

- Organiser un projet complexe avec les Storyboard reference
- Projet et storyboard pour l'iPad ; le composant Split View ;
- La gestion des fenêtre PopOver ; les fenêtres modales (Form, Alert et Actions) ; UIAlertController
- Créer, organiser et positionner des éléments sur la barre à outil

Gérer les différents formats avec les Size classes

- Storyboard universelle ; classe de tailles ; tailles compactes et normales ;
- Notions de traits ; gérer les tailles dans Xcode ; simuler les différentes tailles et orientation ; adapter l'interface en fonction d'une taille particulière

Audience

Développeurs, Chefs de projet ayant des notions de programmation et souhaitant construire des applications iPhone et iPad

Pré-requis

- Notions de programmation ; typiquement Développeurs Java, C#, AS3, C++, C, Delphi, PHP, JavaScript...

- Disposer d'un ordinateur Mac équipé de la dernière version d'OSX

Programme (suite)

Architecture

- Rappel sur les Cocoa Design Patterns
- Créer des bibliothèques (framework)
- Github, git, cocoaPods, carthage
- Bus applicatif NSNotificationCenter
- Proposition d'architecture

Déploiement

- Comprendre le processus de déploiement (TestFlight, AppStore, Entreprise)
- Finalisations : SplashScreen, icônes, descriptifs
- Building : configurations, project settings, targets, scheme
- Comprendre les certificats d'Apple (provisioning, entitlements)
- Aperçu d'iTune Connect

Audience

Développeurs, Chefs de projet ayant des notions de programmation et souhaitant construire des applications iPhone et iPad

Pré-requis

- Notions de programmation ; typiquement Développeurs Java, C#, AS3, C++, C, Dephi, PHP, JavaScript...

- Disposer d'un ordinateur Mac équipé de la dernière version d'OSX

Méthode et Moyens Pédagogiques mis en oeuvre

- Présentations de concepts effectuées par le formateur à l'aide de slides/diapositives transmises depuis l'ordinateur du formateur sur un écran à l'aide d'un video projecteur ;
- Démonstrations effectuées par le formateur sur son ordinateur à l'aide des logiciels présentés dans la formation et retransmis sur un écran par un video projecteur ;
- Exercices guidés, disponibles sur le support des stagiaires et effectués par les stagiaires sur leurs ordinateurs sous le contrôle du formateur ;
- Exercice guidé en fin de journée, reprenant les notions acquises dans la journée réalisé de manière autonome par les stagiaires sur leurs ordinateurs ;
- Résumé du module effectué en fin ou début de journée par le formateur à l'aide de slides/diapositives transmises depuis l'ordinateur du formateur sur un écran à l'aide d'un video projecteur ;
- Test, sous forme de questions à choix multiples, effectué par le formateur, projeté sur un écran, et auquel les stagiaires répondent par oral.