

iOS - Développer des applications iPhone et iPad

4 j (28 heures)

Ref : IPAD

Public

Ingénieur logiciel
Développeur
Analyste programmeur

Pré-requis

Connaissances de la programmation et d'un langage orienté objet (C++, Java, C#, PHP5...)

Moyens pédagogiques

Formation réalisée en présentiel ou à distance selon la formule retenue
Exposés, cas pratiques, synthèse, assistance post-formation pendant trois mois
Un poste par stagiaire, vidéoprojecteur, support de cours fourni à chaque stagiaire

Modalités de suivi et d'évaluation

Feuille de présence émargée par demi-journée par les stagiaires et le formateur
Exercices de mise en pratique ou quiz de connaissances tout au long de la formation permettant de mesurer la progression des stagiaires
Questionnaire d'évaluation de la satisfaction en fin de stage
Auto-évaluation des acquis de la formation par les stagiaires
Attestation de fin de formation

Objectifs

Comprendre comment concevoir une application mobile
Apprendre à développer des applications complexes pour iPhone
Maîtriser l'environnement de développement intégré MacOS-X - XCode
Maîtriser l'environnement du générateur d'interface
Maîtriser l'environnement de iPhone Simulator et du debugger intégré
Être capable d'utiliser les API du SDK pour interagir avec l'iPhone ou l'iPad

Programme détaillé

LES ARCHITECTURES IPHONE / IPOD TOUCH / IPAD

Les besoins en ressources selon l'utilisation
Impacts du firmware sur les applications

Contraintes de développement liées à la plateforme iPhone

Particularités du système d'exploitation iOS

Pourquoi utiliser Objective-C ?

L'ENVIRONNEMENT DE DEVELOPPEMENT

Configuration Mac nécessaire

Présentation des outils et IDE disponibles (Xcode, Interface Builder, iPhone Simulator)

Démarrer un nouveau projet avec Xcode

Compiler à destination de l'iPhone / iPod touch / iPad

Compiler à destination de l'iPhone / iPod touch (Organizer)

La signature du binaire (Target) ?

Les pragma XCode / compilateurs ObjC

Concevoir ses propres modèles (templates) de projet XCode

Le simulateur iPhone

LE LANGAGE OBJECTIVE-C 2.0

Présentation générale et vocabulaire

Objective-C 2.0 runtime référence

Comparatif langages orientés objet Objective-C, C++ et Java

Message d'appel à une méthode

Les méthodes et les classes

Création de classes et d'instances

Les conventions de nommage

Accès et portée des variables d'instances

Propagation des messages

La directive @selector

LA PROGRAMMATION OBJET AVEC OBJECTIVE-C 2.0

Catégories et extensions

Les propriétés déclarées (@property)

L'utilité des protocoles (@protocol)

Déclarations des interfaces donnant la responsabilité à d'autres pour l'implémentation

Une gestion de la mémoire performante

Comprendre et savoir utiliser le « runtime reference »

LES API COCOA TOUCH DU SDK IPHONE

Tour d'horizon des principales API par thème

Les grandes lignes des méthodes de codage avec Cocoa

Classes de base et hiérarchie

LA GESTION DES DONNEES AVEC LE FRAMEWORK FOUNDATION

Philosophie du framework foundation

Les classes fondamentales communes à tous les traitements

Les collections

String, Text & Fonts (NSStringSet)

Dates, Times & Numbers (NSDate)

Interfacer son application avec les réglages utilisateurs standards système

L'INTERFACE UTILISATEUR POUR VOS APPLICATIONS IPHONES

Framework: Cocoa Touch Layer

Le système de coordonnées des Views

Hiérarchies des classes UIResponder / UIWindow / UIView

Combiner une Barre d'onglets" (Tab Bar) et des Contrôleurs de Navigation

Les apports du framework UIKit

Le modèle MVC (Model View Controller)

Les classes Contrôleur de View (Controller View)

UIView, sous-classe UIResponder (réponse/gestion événements utilisateurs)

La hiérarchie des Views et leur géométrie

Les interactions entre les objets View dans UIKit

L'architecture de rendu des Views

Les bases de Core Animation

DISTRIBUER SON APPLICATION

Présentation de la distribution de son application sur l'Appstore

Les « contraintes » imposées par Apple

Le certificat de distribution
