



การเขียนโปรแกรม iOS ขั้นสูง เพื่อรองรับการให้บริการบนอุปกรณ์เคลื่อนที่ (Mobile Devices)



ในปัจจุบันจำนวนแอปพลิเคชันบนมือถือเติบโตขึ้นอย่างรวดเร็ว เพื่อตอบสนองความต้องการใช้งานและการให้บริการซึ่งเปลี่ยนแปลงไปตามชีวิตประจำวันของแต่ละบุคคล โดยสื่ออุปกรณ์ที่ให้บริการประเภท iPhone และ iPad มีจำนวนฐานผู้ใช้งานเป็นจำนวนมาก เป็นเหตุจำเป็นในการพัฒนาแอปพลิเคชันบนมือ

จำเป็นต้องสร้างแอปพลิเคชันในระบบปฏิบัติการIOS เพื่อตอบสนองการใช้งานของผู้ใช้กลุ่มนี้ ดังนั้นจึงมีความจำเป็นที่ต้องอบรมบุคลากรที่ปฏิบัติงานในภารกิจไอที ให้สามารถนำองค์ความรู้ที่ได้จากผู้พัฒนาที่มีความเชี่ยวชาญ สามารถสร้างแอปพลิเคชันที่ใช้งานในระบบปฏิบัติการIOSได้ และสามารถพัฒนาแอปพลิเคชันเพื่อเผยแพร่ข้อมูลอันเป็นประโยชน์แก่ประชาชน

วัตถุประสงค์:

- เพื่อให้ผู้เข้ารับการอบรมมีความรู้ความเข้าใจเกี่ยวกับการพัฒนาเทคนิคการเขียน วิเคราะห์ ออกแบบและสร้างแอปพลิเคชันบนมือถือด้วยระบบปฏิบัติการIOS เพื่อให้ผู้เข้ารับการอบรมมีความเข้าใจโครงสร้างข้อมูลในการเรียกใช้งานผ่านเว็บเซอร์วิส
- เพื่อให้ผู้เข้ารับการอบรมมีความเข้าใจโครงสร้าง ภาษาสวิตซ์ (Swift)
- เพื่อให้ผู้เข้ารับการอบรมเข้าใจส่วนประกอบของ Tool ในการพัฒนาเพื่อให้เกิดความชำนาญเมื่อนำไปใช้งานจริง

กลุ่มเป้าหมาย:

- นักพัฒนาแอปพลิเคชันทั่วไป
- โปรแกรมเมอร์และผู้ดูแลระบบ
- ฝ่ายไอที ผู้ดูแลและพัฒนาระบบขององค์กร
- ครู อาจารย์ ที่ต้องการพัฒนาทักษะและความรู้ในการเขียนโปรแกรม
- เจ้าหน้าที่สำนักเทคโนโลยีสารสนเทศ



ความรู้พื้นฐาน:

- มีความรู้ความเข้าใจเรื่องการติดตั้ง/ถอนการติดตั้ง โปรแกรม
- มีความรู้ในการใช้อินเตอร์เน็ตและเครือข่ายพื้นฐาน
- ใช้งานอินเทอร์เน็ตและระบบปฏิบัติการวินโดวส์ได้
- เคยเขียนโปรแกรมหรือเคยผ่านการอบรมภาษา C มาบ้างเล็กน้อย

ระยะเวลาในการอบรม:

- 30 ชั่วโมง (5 วัน)

ราคาคอร์สอบรม:

- 17,500 บาท / คน (ราคานี้ยังไม่ได้รวมภาษีมูลค่าเพิ่ม)

วิทยากรผู้สอน:

- อาจารย์สามิตร โกยม และทีมงาน

คอร์สที่ควรอบรมก่อนหน้า:

- Basic iOS Development

เนื้อหาการอบรม:

Training includes

Understand the language used to develop applications function perfectly and started building apps based on function.

1. Learn the basics of the language and tools.

- Installation development tools like Xcode.
- Programming Object Oriented Programming (OOP).
- Programming with Swift 2.0 (for iOS).



- The difference between Swift with Objective-C.
- Make Sytax and API based Swift.

2. Swift started writing through various gimmicks. Need to Know

- Naming Class and Function
- NSObject known and Syntax in succession (Inheritance).
- Creating objects in Swift and Obj-C.
- Creating Default and Custom Constructors in Swift.
- About Data-Type needed Scalar Data-Type and Pointer Data-Type.
- Mutable known and Non-Mutable Data Collection.
- About ID Data-Type and writing Polymorphism.
- Categories with the Method Injection, an advanced technique.
- Dynamic Data Type Verification

3. Development of Apple iOS with English Swift.

- Syntax basics such as Variable, function types.
- The announcement Class, Protocol, Extension.
- The API uses such as Array, Dictionary, String, println and more.
- Class inheritance Class Inheritance
- The init constructor and custom constructor.
- Creating a Bridging-Header Files for use in communication between Swift and Obj-C.

Understand the functioning of Cocoa Touch and fully Connected to the Native Database.

4. Start writing iPhone Application

- Xcode 7.x taught using the latest version.
- Creating projects and editing projects.
- Learn UIApplication Flow that is triggered when an application was opened.
- Learn to create applications and computer applications, starting with the empty project in order to understand all the elements.



- Learn the Model-View-Controller (MVC).
- Sandbox-Folder structure and the interior of the iPhone.
- Try doing that with workshops explaining the various relationships.
- UIApplication, AppDelegate, Window, UI View.
- ViewController
- Storyboard, Navigation Controller, FileOwner
- IBOutlet and IBAction
- Creating a file XIB, Storyboard to create a User Interface up a manual.
- Writing applications and applications without file XIB (many cases need to write yourself).

5. Drill building and operating of. UIViewController And the sequence of events known Delegate

Methods conditions.

- The screen will not load.
- The screen will appear, the screen displayed.
- The screen is gone, the screen is gone.
- Conditions of the Low Memory Warning and writing application functions to support.

6. Using Widgets and all matters relating to the UI, for example.

- Animation for the app offers even more attractive look.
- Management UIView to support screen rotation.
- Use IBAction and IBOutlet advanced.
- Creating a Custom Table Row UITableView and professional.
- Creating UINavigationController and UIBarButtonItem
- Loading attached to the Application NSBundle.



7. persistent data management (Persistent Storage) Advanced (Dynamic and Bundle).

- (Property List) File .plist
- (Relational Database) Sqlite3 which is compatible with Android.
- The database created from the outside, both command-line and GUI SQLite Manager.
- Learning problems that often occur with SQLite as Thailand Language Encoding.

8. Management UI with large-screen iPhone 3.5 ", 4", 4.7 ", 5.5" "(iPhone 3,4,5,6,6+), for example.

- The recommended bundle font free fonts and beautiful.
- Capture screen size
- To make the app function supports both conventional screen, retina 3.5 "and retina 4".
- Conditions of the Low Memory Warning and programming to support this state.
- Customizing the UI to deal with iOS7 and 8 perfectly.

Taught Advanced features in Development of Connectivity and Applications with Advanced

Network.

9. Multi-screen as well. TabBarController and UINavigationController

- Installation UINavigationController different either through the storyboard and programmatic.
- Creating such a UIBarButtonItem Reusable and Custom.
- Understanding Changing Screen Normal and a Modal UIViewController.
- Segue to use to trap the transition between the screen and the screen. ViewControllers

10. Using Gesture different patterns to trap the touch screen Tap, Pitch, Rotate.

- Writing Private API and caution in order not to hit the Apple Private API Rejected.
- Writing apps functionality to support multiple languages with a technique of Localization.
- Write an application to connect to the Internet with AFNetworking Library.



11. The Webserver on OS X.

- Writing code to look and Professional with Pragma Preprocessor.
- The written application form Multi-Tasking Custom Threading with the modern division of labor can have multiple Core-CPU's.
- Advanced Thread Handling - Communication between Main and Custom Thread.
- Basic Image Processing Techniques example.
- Compression Quality Compression
- Designed to accommodate a screen Retina Display.

12. Fetch a mySQL Database Server with the Format in JSON, XML and Installation of Native Email, Twitter and Facebook and Connect

Webservice (lectured extensively both OBJ-C and SWIFT).

13. Reading XML data using (Advanced Parser).

- Technical Walk Through
- Capture XML Start Tag, End Tag and Found Character.
- Applications for the Development of Web Map Service and Advanced.

14. The development of an application form. Location-Based System which is compatible with GPS and Map.

- Fetch coordinates with the Core Location Framework of Location Tracking System with GPS, Wi-Fi.
- Know the limitations of the iOS Simulator to simulate and coordinates solution.
- Conditions detection coordinate changes.

15. iPhone programming interface Webservice both Soap and Restful.

- Webservice connection as a self-sign SSL or untrusted HTTPS.



16. Install Apps on Maps application with GoogleMap SDK.

- Getting Google Service API Key
- Enabling users to interact with the map.
- The coordinates of the device via Wi-Fi and GPS.
- Pane or Zoom principle to the (local) place.
- The Advanced Zoom to aim it at one point, the group coordinates appropriately.
- Creating a Place Marker Annotation or that can interact with users.
- Search by name, location coordinates and Geocoding search by name, location coordinates
Reverse Geocoding.
- Draw a path with Google Direction API.
- Draw and customize the polygon MapView.

17. Understanding the Universal Storyboard to manage multiple screen sizes.

- Constraints on the techniques of making Autolayout efficiency.
- The problems that often arise in the Autolayout.

18. Prepare drug with everything up before Submit Appstore.

- Submit Application for the iPhone Appstore
- The Configure File Info.plist Submit valid before the AppStore.
- iPhone Developer Program subscription model.
- Creating a signing key, digital certificate to use in the process of installing applications or device.
- Defined app and device ID in iTunesConnect.
- App icon design techniques and tools in the UI design efficiency.