



# Mobile Application Development with iOS

All popular apps are created for Android and iOS devices, and some for Windows too. This is mainly because developers want maximum outreach for their apps and millions of people use iPhones all over the world. So, if you are an aspiring app developer, iOS app development will definitely put you among the major league of developers!

## Why use iOS for App Development?

- Apple's IDE (Integrated Development Environment) for Mac as well as iOS is Xcode. Xcode is essentially a graphical interface used to write apps. That along with the programming languages such as Swift or Objective-C is used when developing apps for iOS.
- Apple has high standards for their apps, which pushes the programmer to develop excellent quality apps.
- Sure, the number of downloads of Android apps are higher than the iOS, but the revenue generated by the in-app sales of iOS outscores Android by a large margin.

## Why you must learn iOS App Development?

- If you want to become an iOS app developer or a mobile app developer, in general, learning iOS will be a huge advantage to get you started.
- Apple apps have a certain set of criteria that need to be fulfilled. Knowing what they are will give you better clarity about developing apps for iOS devices in the future.
- App development is an excellent skill to have as many businesses are trying to solve the problems of their existing or future customers through apps. And the graph is only going higher.
- Here's the big one...Since the launch of App Store in 2008, more than 1,530,000 jobs have been created by the US

## Course Duration:

- 1 Month

## Course Syllabus:

### 1: Introduction

- iPhone and iPad Device Anatomy
- iOS Architecture and SDK Frameworks
- iOS and SDK Version Compatibility
- Apple iOS Developer Program

### 2: Xcode 5

- Tour of the IDE
- Templates, Projects, and Workspaces
- Creating a New Project
- LLVM and LLDB
- Debug Gauges
- Asset Management
- XCTest Testing Framework
- Continuous Integration and Bots
- Automatic Configuration

### 3: Objective-C for Experienced Programmers

- Classes, Objects, and Methods
- Declared Properties
- Memory Management
- Automatic Reference Counting (ARC)
- Categories and Extensions
- Formal and Informal Protocols
- Blocks
- Application Patterns and Architecture
- Model View Controller (MVC)
- IBOutlets and IBActions
- Subclassing and Delegation

### 4: Views and Windows

- The View Hierarchy
- Containers
- Controls

- Text and Web Views
- Navigation View and Tab Bars
- Alert Views and Action Sheets
- Controlling Rotation Behavior
- View Autosizing
- Autolayout

## 5: Storyboards

- Adding Scenes
- Segues
- Transitions
- Using in a Tab Bar Application

## 6: Table Views

- Static and Dynamic Table Views
- Delegates and DataSources
- Table View Styles
- Custom Cells

## 7: Navigation Based Applications

- Adding the Root View Controller
- Creating the Navigation Controller
- Controlling the Stack Navigation Programmatically

## 8: UIPickerView and UIDatePicker

- Designing the UI
- Coding for the Data Picker
- Hiding the Keyboard
- Memory Management

## 9: Directories and Files

- NSFileManager, NSFileHandle, and NSData
- Problems Solved by ADO.NET Entity Framework
- Pathnames in Objective-C
- Working with Directories
- Working with Files
- Reading and Writing from a File
- iCloud

- Key-Value Data
- Archiving

## 10: Working with Data

- SQLite Integration
- Using SQLite Directly
- Overview of Core Data
- Managed Objects
- Persistent Store Coordinator
- Entity Descriptions
- Retrieving and Modifying Data

## 11: Multitouch, Taps, and Gestures

- The Responder Chain
- Touch Notification Methods
- Enabling Multitouch on the View
- Gesture Motions
- Gesture Recognizers

## 12. Drawing EXPERIENCE BEYOND IT TRAINING

- Core Graphics and Quartz 2D
- Lines, Paths, and Shapes

## 13: Animation

- Core Animation Blocks
- Animation Curves
- Transformations

## 14: Multitasking

- Application States
- Background Execution
- Background App Refresh in iOS 7
- State Restoration

## 15: Notifications

- Local Notifications
- Push Notifications

## 16: Core Location Framework

- Location Accuracy
- Obtaining Location Information
- Calculating Distances
- MapKit Framework and MKMapView

## 17: Concurrency

- Grand Central Dispatch (GCD)
- Serial and Concurrent Queues
- Main Dispatch Queue
- Completion Blocks
- Operation Queues

## 18: Networking

- Reachability
- Synchronous Downloads
- Asynchronous Downloads
- Handling Timeouts
- Sending HTTP GET and POST Requests
- Parsing JSON
- Parsing XML
- AirDrop

## 19: Targeting Multiple Devices

- iPhone vs. iPad
- Universal Apps
- Multiple SDK Support
- Detecting Device Capabilities
- Supporting iOS 6 and iOS 7

## 20:Localization

- Resources
- Language and Region
- NSLocale
- Text
- Dates
- Numbers
- Performance and Power Optimization
- Measuring Performance
- Instruments
- Responsiveness
- Memory Usage, Spikes, and Leaks
- Networking and Power

