

IOS Mobile Application Development

Duration: 8 days / 64 hours

Prerequisites: Working Knowledge of Any OOP Programming Language

Day 1: Introduction to iOS & Swift Programming Basics

Topics Covered:

- Introduction to iOS ecosystem and tools
 - Xcode and iOS Simulator
 - iOS Architecture Overview
- Introduction to Swift Programming
 - Constants and Variables
 - Data Types
 - Optionals
 - Control Flow (if, switch, loops)

Hands-on Labs:

- Set up Xcode and create your first iOS app
 - Simple calculator using Swift
-

Day 2: Deep Dive into Swift Language

Topics Covered:

- Functions and Closures
- Enumerations and Structures
- Classes and Objects
- Protocols and Extensions
- Error Handling in Swift

Hands-on Labs:

- Create a custom class with properties and methods
 - Using protocol and delegation pattern in a mini-app
-

Day 3: iOS UI Development with SwiftUI - Basics

Topics Covered:

- Introduction to SwiftUI
- Views, Modifiers, and Layout System
- State and Binding
- Navigation and Stacks

- List and Form elements

Hands-on Labs:

- Build a to-do list app using SwiftUI
 - UI Design using VStack, HStack, and ZStack
-

Day 4: Advanced SwiftUI & Accessibility

Topics Covered:

- ObservedObject, EnvironmentObject, StateObject
- Animations and Transitions in SwiftUI
- Introduction to Accessibility
 - VoiceOver
 - Dynamic Type
 - Accessibility Inspector

Hands-on Labs:

- Refactor To-Do app with data binding
 - Implement accessibility labels and traits
-

Day 5: Networking with REST API

Topics Covered:

- URLSession and Codable
- Making GET & POST requests
- Parsing JSON data
- Error Handling in Network Requests

Hands-on Labs:

- Connect app with a public REST API
 - Display dynamic data in a list view
-

Day 6: Design Patterns in iOS Development

Topics Covered:

- Builder Pattern
- Adapter Pattern
- MVVM (Model-View-ViewModel)
- Clean Architecture Principles

Hands-on Labs:

- Implement MVVM in a weather app
- Adapter pattern example for image downloader

Day 7: Modern Concurrency in Swift

Topics Covered:

- Introduction to Concurrency
- Grand Central Dispatch (GCD)
- async/await in Swift
- Task and TaskGroup
- Structured Concurrency

Hands-on Labs:

- Use async/await to fetch and display data
 - Parallel image downloads using TaskGroup
-

Day 8: Writing Unit Tests and Final Project

Topics Covered:

- XCTest framework overview
- Writing unit tests for Swift classes
- UI testing basics
- Code coverage

Hands-on Labs:

- Write unit tests for API calls and business logic
- Final Capstone Project: A complete app implementing UI, API, and MVVM with tests