

iOS Application Development

iOS-101

Duration: 4 days; Instructor-led

Time: 9:00 AM – 5:00 PM

ABOUT THIS COURSE

This course is an introduction to iOS application development. It covers how to create and compile projects using Xcode, basics of the Swift language and uploading the completed app on the Apple App Store.

During this course, you will create a reasonably complex iOS program. In the process, it covers storyboards, table view, table view controller, editing table view, navigation controllers, camera, auto layout, gesture recognisers, push view controllers and modal view controllers. You will also learn how to use unit tests and debugging tools.

OBJECTIVES

Upon completion of this program, participants should be able to:

- Create a reasonably complex iOS application, using storyboards, table view, table view controller, navigation controllers, camera, autolayout, gesture recognisers, push view controllers and modal view controllers.
- Test an application using unit tests and debugging tools.
- Save application data so it persists between runs.

PREREQUISITES

Knowledge of a programming language is recommended.

AUDIENCE

This program is targeted for experienced developers with no programming experience in iOS Development

SYSTEM REQUIREMENTS

- A Mac computer running macOS 10.15 Catalina or 11.0 Big Sur
- Xcode 12.x
- Swift 5.x

PROGRAMS YOU WILL WRITE

Food Tracker - A simple meal tracking app. This app shows a list of meals, including a meal name, rating, and photo. A user can add a new meal, and remove or edit an existing meal. To add a new meal or edit an existing one, users navigate to a different screen where they can specify a name, rating, and photo for a particular meal.

COURSE CONTENTS

Module 1: A Swift Tour

- Simple values
- Control Flow
- Functions and closures
- Objects and classes
- Enumerations and structures

- Protocols and Extensions
- Error handling

Module 2: Build A Basic UI

- Create a project in Xcode
- Identify the function of key files that are created with an Xcode project template
- Open and switch between files in a project
- Run an app in Simulator
- Add, move, and resize UI elements in a storyboard
- Edit the attributes of UI elements in a storyboard using the Attributes inspector
- View and rearrange UI elements using the outline view
- Preview a storyboard UI using the Preview assistant editor
- Lay out a UI that automatically adapts to the user's device size using Auto Layout

Module 3: Connect The UI To Code

- Explain the relationship between a scene in a storyboard and the underlying view controller
- Create outlet and action connections between UI elements in a storyboard and source code
- Process user input from a text field and display the result in the UI
- Make a class conform to a protocol
- Understand the delegation pattern
- Follow the target-action pattern when designing app architecture

Module 4: Work With View Controllers

- Understand the view controller life cycle and when its callbacks occurs, such as viewDidLoad, viewWillAppear and viewDidAppear
- Pass data between view controllers
- Dismiss a view controller
- Use gesture recognizers as an additional level of generating events
- Anticipate object behavior based on the UIView/UIControl class hierarchy
- Use the asset catalog to add image assets to a project

Module 5: Implement A Custom Control

- Create and associate custom source code files with elements in a storyboard
- Define a custom class
- Implement an initializer on a custom class
- Use UIView as a container
- Understand how to display views programmatically

Module 6: Define Your Data Model

- Create a data model
- Write failable initializers on a custom class
- Demonstrate a conceptual understanding of the difference between failable and nonfailable initializers
- Test a data model by writing and running unit tests

Module 7: Create A Table View

- Understand the key components of a table view
- Create and design a custom table view cell
- Understand the roles of table view delegates and data sources
- Use an array to store and work with data
- Display dynamic data in a table view

Module 8: Implement Navigation

- Embed an existing view controller within a navigation controller in a storyboard
- Create segues between view controllers
- Edit the attributes of a segue in a storyboard using the Attributes inspector
- Pass data between view controllers using the `prepareForSegue(_:sender:)` method
- Perform an unwind segue
- Use stack views to create robust, flexible layouts

Module 9: Implement Edit And Delete Behaviour

- Differentiate between push and modal navigation
- Dismiss view controllers based on their presentation style
- Understand when to use different type cast operators for downcasting
- Leverage optional binding to check for complex conditions
- Use segue identifiers to determine which segue is occurring

Module 10: Persist Data

- Create a structure
- Understand the difference between static properties and instance properties
- Use the NSCoding protocol to read and write data

Module 11: Working with the Web, JSON and Concurrency

- The basics
- Create a URL
- Create and Execute a network request
- Process the Response
- Work with an API
- Modify a URL with URL Components
- JSON Basics
- Decoding into Swift Types
- Convert JSON Data to Swift Types
- Decoding into Custom Model Objects
- Update the Request Completion Handler
- Where to put your code
- Write a Completion Handler
- Addressing Failure
- Decide where the Function should live
- Concurrency and Grand Central Dispatch
- Fetch and display the photo
- App Transport Security and the HTTP Protocol

Assignment

Student will design and build a simple application