

iOS Mobile App Development with Swift

Duration:

5 Days

What is the course about?

In this course you will learn how to develop professional iOS mobile apps using the Swift programming language. iOS mobile apps have been primarily developed using Objective-C prior to the introduction of the Swift programming language. Apple introduced a new programming language called Swift as a way to reduce the entry level requirements into programming iOS apps. In this course you will develop mini-apps to introduce the different iOS concepts. You will develop a Chuck Norris joke app that pulls random Chuck Norris jokes from an API. You will also develop a movies reviews app that connects to an API and pulls movie reviews, popular movies, and movies that are currently showing. As the course progresses you will also develop your own app based on your own idea using the concepts introduced in the course. You will be required to take what you learn from the course and apply the use the concepts introduced in class and incorporate them into your own app.

Duration

This is a full time course offered over 5 days. By the end of the course the delegates would have created a fully working iOS mobile app.

Programming Experience

Swift is an easy language to learn and reduces the barrier to developing iOS mobile apps, however, a good command of object oriented programming concepts is required in order to succeed in this course.

Technical Skill

The Mac OS X environment will be used for developing iOS mobile apps. A working knowledge of using a Mac or the ability to pick up the Mac OS X environment will be required.

Private Training

We offer the training privately to a team, group or a company. The training runs over 5 days. The course curriculum can be customized to your needs. Training can be conducted on your premises or our premises. The course price onsite is R9 500 per delegate and R12 599 to run it on our premises. Training can be conducted on our premises in either Johannesburg, Durban or Cape Town. We work with you to find suitable dates for your team as there is no fixed start date for the course.

Public Training

We also conduct the training publicly. This is suitable for individuals or delegates that cannot book the private training. The class will be comprised of delegates and individuals from

different companies. The training content and duration is the same and training will run at our premises in Johannesburg or Cape Town, South Africa. The course price is R12 599 per delegate. A minimum of 4 delegates is required to run the course. A preliminary date is scheduled and will be confirmed once we have 4 delegates to run the course.

Course Topics

App Fundamentals

We'll setup our environment, walk through the basic architecture of an app, and put together a simple UI with Interface.

Setup:

- Setting Up your Environment
- Starting a new Xcode Project

App Structure:

- Understanding an Xcode Project
- App Architecture

Project - Flicks Client

The purpose of this project is to develop a basic MVC application and start playing with table views, one of the more complex, but important views in iOS.

Views and View Controllars

We'll continue to focus on views and view controllers. We'll cover navigation between view controllers, events, and responsive layout using Auto Layout.

- Auto Layout
- Views
- View Programming Guide
- Table Views
- Navigation in iOS
- View Controllers
- View Controller Programming Guide
- View Controller Lifecycle
- Passing Data Between View Controllers
- Forms and Events

Networking and Authentication

We'll learn about networking and authentication, specifically for RESTful APIs. It's common to use networking libraries when interacting with RESTful APIs to handle tasks such as connection pooling, setting HTTP headers, serialization/deserialization, and authentication. We will also recommend some design patterns for implementing model classes.

- Views
- Pull-to-refresh
- Networking
- OAuth 1.0a Overview
- A Conceptual Look at OAuth

- Managing signed-in user
- Auto Layout
- Configuring constraints for adjacent UILabels
- Dynamically modifying Auto Layout constraints

Animations, Gestures and Container View Controllers

We'll cover some more advanced topics for views and view controllers. We'll play with view animations, as well as attaching and handling gesture recognizers. For view controllers, we'll create our own custom container view controllers.

- Views
- View Animation Overview
- Using Gesture Recognizers
- Moving and Transforming Views With Gestures
- View Controllers
- Custom Container View Controllers - Quickstart
- Custom Container View Controllers - Guide

Device Frameworks

We'll explore the many iOS device frameworks, including location, camera, maps, push notifications, and accessing data on the phone.

- Device Frameworks
- Location
- Maps
- Camera
- Apple Docs: AV Foundation Overview
- Parse Guide: Push Notification

Multithreading and Persistence

We'll overview the various persistence options: NSUserDefaults, the file system, SQLite, and Core Data.

Persistence

- NSUserDefaults
- File System
- SQLite
- Core Data

Multithreading

- NSOperation and NSOperationQueue
- Grand Central Dispatch
- NSTimer

UIKit Dynamics

The goal of this week is to explore UIKit Dynamics, the physics engine released in iOS 7. People have an intuitive and subconscious understanding of physics: how a ball will bounce against a wall or how a rubber band snaps. Bringing physics into your apps will give people the illusion that they are manipulating physical, not digital objects.

We will explore how to assemble various "behaviors" such as gravity, collision, and push to create different physical effects.

- UIKitDynamics

Gravity

- Add and remove gravity to objects
- Set the direction and force of gravity

Collision

- Cause objects to collide with each other
- Cause objects to collide with a boundary
- Detect when an object has collided with an item or boundary

Push

- Apply a continuous or instantaneous force to an object
- Set the direction and force of the push