Полная программа

iOS Developer. Professional

iOS Developer. Professional

Длительность курса: 144 часа

Модуль 1. SwiftUI и основы Combine

Тема 1
Проектируем UI декларативно с SwiftUI. В чем отличия UIKit и SwiftUI

Цель занятия
настроить окружение для работы на курсе и выполнения домашних заданий;
использовать Xcode;
создавать базовые интерфейсы на SwiftUI/Combine.

Краткое содержание
возможности и ограничения SwiftUI;
property Wrappers: @State, @Binding, @ObservedObject и ObservableObject, @EnvironmentObject, @Environment;
основы Combine: @Published;
решение организационных вопросов;
настройка environment: Xcode, git, scripts.

Домашние задания
Создание каркаса приложения на SwiftUI
Цель
1. Целостно понимать навигационный стек SwiftUI/Combine.
2. Получить умение сборки иерархии экранов на SwiftUI/Combine.

Тема 2
SwiftUI List, List с кастомным лейаутом, Hosting ViewControllers

Цель занятия
понимать, в каком состоянии приложение;
добавлять логику на изменение состояния приложения;
знать как устроен UIKit и его иерархия классов;
делать навигацию в SwiftUI разными способами.

Краткое содержание
Life cycle приложения;
UIResponder;
SceneDelegate;
Hosting ViewControllers и зачем они;
Combine.

Тема 3
Использование NavigationView, TabView. Нативная навигация, бекпорты и фреймворки для поддержки всех функций

Цель занятия
использовать NavigationView и TabView в SwiftUI;
использовать back ports и сторонних решений;
организация deeplinks и тестирования навигации.

Краткое содержание
особенности навигации в SwiftUI;
выбор фреймворков для решения проблем с навигацией;
поддержка iOS 14-15;
открытие экранов через механизм Deeplinks;
небольшой юнит тест определения навигационных действий.

Тема 4
Создание кастомных Shape, SwiftUI Drawing and Animation API

Цель занятия
работать с CALayer и понимать систему координат, используемую в CoreGraphics;
программно рисовать Shapes в SwiftUI;
использовать анимацию в SwiftUI.

Краткое содержание
разница CoreGraphics и Drawing API SwiftUI;
создание кастомных фигур на Drawing API SwiftUI;
анимация интерфейса в SwiftUI.

Домашние задания
Реализация пейджинга на реальном API
Цель
Создать список с пейджингом, работающий на реальном серверном API.

Тема 5
Отображение структурированных данных, List, пейджинг, кастомные компоненты на UIViewRepresentable, принципы HIG

Цель занятия
реализовывать пейджинг на SwiftUI;
кодогенерить Network слой в ДЗ;
привязывать List к реальному API с помощью кодогенерации.

Краткое содержание
использование List в SwiftUI;
реализация пейджинга для List.

Модуль 2. Современная архитектура мобильных приложений

Тема 1
Необычная система типов Swift, структуры данных, Generics, Связывание разных частей приложения Observing, Signals, Callbacks. PATs (Protocol with Associated Types)

Цель занятия
создать кастомные структуры данных.

Краткое содержание
теория типов и Compound и Named типы;
метатип и вложенные типы;
protocol Composition;
generics.

Тема 2
Protocol Oriented Programming (POP), SOA, модуляризация, архитектурные Rx паттерны: Flux/Redux

Цель занятия
применять паттерн Inversion of Control;
использовать ServiceLocator;
объяснить в чем польза слоистой архитектуры.

Краткое содержание
inversion of Control паттерн;
serviceLocator и инжектинг;
разделение архитектуры на слои и что это дает;
модуляризация приложения, способы: Frameworks, SwiftPM;
protocol Oriented Programming (POP);
type constraints.

Домашние задания
Реализация Service Locator и Dependency Injection и изолированности слоев в приложении
Цель
Научиться внедрять DI, понять плюсы подхода.

Тема 3
MVx, VIP, анализ архитектурных паттернов. SOLID, GRASP, Coupling/Cohesion принципы, DI.

Цель занятия
разобраться в семействе MV(x) паттернов;
объяснить, что такое модуляризация и как ей пользоваться;
рассказать о паттерне Clean Architecture подход;
использовать паттерны в мобильной разработке.

Краткое содержание
MV(X) архитектурные паттерны;
архитектурные Rx паттерны;
SOLID и как получить от него пользу;
KISS, DRY/DIE, YAGNI, BDUF, SDC;
необходимые паттерны для сегодняшней мобильной разработки: Adapter, Memento, Observer, Strategy, Command, Composite, Iterator, Mediator, Proxy, Template Method, Singleton и где они применены в iOS.

Домашние задания
Создание модуляризированного приложения с SOA/MVVM/Flux/MobX
Цель
Прокачать умение делать архитектурный рефакторинг всего приложения.
Научиться внедрять Архитектурные паттерны, понять плюсы подхода.

Модуль 3. Foundation без сторонних фреймвоков и Swift 5 Standard Library

Тема 1
Swift Runtime, компилятор LLVM, продвинутая отладка с помощью LLDB, скриптинг

Цель занятия
использовать LLVM: AST, SIL, IR;
знать Method Dispatch;
знать особенности компиляции;
использовать LLDB.

Краткое содержание
как работает LLVM: AST, SIL, IR;
ускорение работы приложения с помощью Method Dispatch;
особенности и хитрости компиляции;
создание и использование кастомных команд в LLDB.

Тема 2
Sequences и коллекции, асимптотический анализ: O(1), O(N), O(N*log(N)), O(n^2)

Цель занятия
использовать Sequence и Collection для реализации собственных структур данных;
знать как работает lazy collection, будете понимать концепцию type-erased.

Краткое содержание
sequence и IteratorProtocol;
type-erased типы: AnySequence, AnyIterator, AnyCollection;
lazy Wrappers;
wrappers for Algorithms;
асимптотический анализ встроенных и кастомных структур данных.

Тема 3
String: устройство, конвертация с другими типами, регулярные выражения Swift 5.7

Цель занятия
углубиться в работу строковых данных в Swift;
научиться использовать регулярные выражения для поиска строк или получения подстрок.

Краткое содержание
поговорим о кодировках, о том как получать коды и из чего состоят символы строк;
рассмотрим способы конвертации строк с помощью Expressible, об устройстве литералов;
затронем тему StringInterpolation строк и вывод объектов под debug в консоль.

вторая часть урока будет посвящена работе с регулярными выражениями, рассмотрим устаревший подход NSRegularExpression и новый Regex и RegexBuilder.

Тема 4
Поддержка многоязычности: локализация строк, новый формат XCode 15, кодогенерация и сервисы переводов. Региональные отличия: переводы дат, величин, оцифровывание текста

Цель занятия
научиться делать приложения для экспорта, доступные для иностранных пользователей, чтобы присущие привычки для их местности были учтены;
использовать лучшие практики получения качественного кода с использованием кодогенерации и линтинга ресурсных строк.

Краткое содержание
как завести локализацию строк в проект, какие проблемы могут встретится при разработке и тестировании приложения, как решаются инструментами кодогенерации SwiftGen, R.swift;
поговорим о плюралах и о том как их настроить для строк;
обсудим инструмент Localizer как решение позволяющее автоматизировать редактирование строк. Рассмотрим способы и платформы для профессиональных переводов строк. Под конец углубимся в реализацию для учета региональных различий, рассмотрим работу форматеров DateFormatter, NumberFormatter, MeasurementFormatter и других.

Домашние задания
Реализация кастомной структуры данных (Suffix Array)
Цель
Вы научитесь создавать кастомные структуры данных на основе протоколов Sequence и IteratorProtocol и решать с помощью них реальные задачи в приложениях.

Модуль 4. Многозадачность

Тема 1
Проблемы многозадачности и способы их решения, GCD

Цель занятия
использовать GCD: QoS, Queues, Main Queue и Main Thread.

Краткое содержание
антипаттерны и проблемы: Priority Inversion, Race condition, Deadlock, Resource contention, Starvation, Non-deterministic and Fairness.

Тема 2
Проблемы многозадачности, GCD(libdispatch), OperationQueue

Цель занятия
знать антипаттерны и проблемы: Priority Inversion, Race condition, Deadlock, Resource contention, Starvation, Non-deterministic and Fairness;
избегать антипаттернов;
пользоваться средствами GCD.

Краткое содержание
плюсы и минусы OperationQueue;
внутренности libdispatch: пул тредов, continuation, QoS и как знание это лучше использовать.

Тема 3
Async/Await, Tasks, Groups

Цель занятия
ориентироваться в структурированной многопоточности;
уметь работать с тасками и группами;
использовать инструменты для параллельного запуска задач.

Краткое содержание
структурирование многопоточности с помощью Async/Await;
изолирование многопоточного контекста с помощью Task;
распараллеливание вычислений с помощью Task groups.

Тема 4
Actor, @MainActor, Global actors, Distributed actor (пример по сокетам)

Цель занятия
иметь полное представление о структурированной многопоточности в Swift;
знать как использовать Actors в Swift.

Краткое содержание
Actors;
RunLoop и чем сегодня он может нам быть полезен;
Виды локов;
Global и @MainActor.

Домашние задания
Реализация асинхронного выполнения задач и оценка эффективности подхода
Цель
Научиться внедрять сервис очереди в существующую инфраструктуру приложения, развивая навык рефакторинга для UI кода приложения.

Модуль 5. Networking и хранение данных

Тема 1
Передача данных, Transferable протокол, Bluetooth

Цель занятия
работать с UTType и Transferable;
работать с CoreBluetooth;
знать Multipeer Connectivity.

Краткое содержание
передача данных;
transferable протокол;
работа с Bluetooth внутри приложения.

Тема 2
GraphQL, кодогенерация, отправка и прием сообщений

Цель занятия
попробовать разные способы формирования работы с URL;
объяснить, как работать с чистой URLSession;
познакомиться с GraphQL и как его использовать на iOS.

Краткое содержание
network фреймвок, HTTP, REST, Sockets, GraphQL;
URLSession;
сериализация и десериализация с помощью Codable;
немного о gRPC и кодогенерации.

Тема 3
gRPC, кодогенерация, отправка и прием сообщений

Цель занятия
изучить Protocol Buffers;
начать использовать gRPC.

Краткое содержание
Protocol Buffers и его преимущества для создания эффективного сетевого слоя в iOS приложениях;
рассмотрим примеры использования Google RPC и узнаем, как генерировать код с помощью protobuf-компилятора для iOS;
получим полностью функциональное и эффективное взаимодействие между клиентом и сервером;

обсудим особенности работы с Protocol Buffers и gRPC в iOS, обработку ошибок и асинхронное выполнение запросов;
полученные знания и навыки помогут нам создавать более эффективные и надежные сетевые слои в наших iOS приложениях.

Тема 4
Socket.io, WebSocket и другие сокеты для чатов и мгновенных обновлений, APNs

Цель занятия
разбираться в socket.io, WebSocket и других сокетах для чатов и мгновенных обновлений.

Краткое содержание
мессенджеры;
пуши;
практика вместе с преподавателем.

Домашние задания
Использование локальной(mesh) сети для создания приложений
Цель
Научится использовать Multipeer Connectivity в приложениях

Тема 5
SwiftData

Цель занятия
разобрать строение файловой системы iOS;
познакомиться с некоторыми способами кэширования.

Краткое содержание
виды кеширования.

Тема 6
Безопасность: AppSecurity, Keychain и обфускация API ключей, SSL Pinning

Цель занятия
AppSecurity;
сохранять важные данные в Keychain;
скрывать данные от декомпиляции ipa файла;
защищать приложение от Man-In-The-Middle аттаки.

Краткое содержание
методы обфускации;
Keychain;
SSL Pinning.

Домашние задания
Реализация поддержки оффлайн режима в приложении
Цель
Научится сохранять Codable структуры в файлы, реализовывать кэш.

Модуль 6. Создание приложений для Apple Watch, TV, Mac

Тема 1
Виджеты, AppClips

Цель занятия
делать свои виджеты;
создавать изменяемые виджеты с своим дизайном;
добавлять виджеты к уже существующему приложению.

Краткое содержание
узнаем о возможностях создания iOS Widgets на SwiftUI и App Clips;
изучим, как создавать виджеты для отображения избранного контента и функций приложения на главном экране устройства;
будет представлена информация о настройке виджетов, их разных типах и способах их использования для облегчения доступа к информации или функциям приложения;
будут рассмотрены примеры реализации iOS Widgets и App Clips для обеспечения удобства использования iOS-приложений.

Тема 2
watchOS

Цель занятия
создать приложение для Apple Watch.

Краткое содержание
отличия SwiftUI для watchOS;
интеграция с iOS приложением.

Тема 3
tvOS

Цель занятия
создать приложение для Apple TV.

Краткое содержание
отличия SwiftUI для tvOS.

Тема 4
Shortcuts и Siri интеграция с приложением

Цель занятия
использовать Actions, Siri Intents.

Краткое содержание
дефолтные actions;
создание Actions;
SiriKir Intents.

Домашние задания
Приложение работающее на всех платформах Apple
Цель
Прокачать навык apple-мультиплатформенной разработки

Тема 5
visionOS

Цель занятия
понимать стек разработки под visionOS;
использовать RealityKit для visionOS;
использовать Reality Compose;
знать как собирать базовый UI под visionOS.

Краткое содержание
отличия платформы visionOS от iOS/iPadOS;
SwiftUI для visionOS: ImmersiveSpace, WindowGroup, Model3D;
RealityKit для visionOS;
редактор Reality Compose и Universal Scene Description.

Домашние задания
Приложение для visionOS
Цель
Научится базовой разработке под visionOS

Модуль 7. CoreML и Vision, нейронные сети и машинное обучение

Тема 1
CoreML, CreateML, Vision использование обученных моделей нейронных сетей

Цель занятия
разобраться как устроен CoreML 3+;
использовать готовые обученные модели.

Краткое содержание
CoreML разных версий отличия от нейронок;
практическая демонстрация возможностей.

Тема 2
Обработка видеопотока с помощью CoreML в режиме реального времени с AVFoundation, CoreImage и Metal

Цель занятия
объяснить как создавать сегментацию в реальном времени.

Краткое содержание
алгоритмы ML;
CoreML + Metal, AVFoundation, CoreImage (практическое дело использования модели для видео).

Тема 3
ARKit и RealityKit, LiDAR датчик

Цель занятия
использовать ARKit, RealityKit, LiDAR.

Краткое содержание
ARKit трекинг объектов;
RealityKit создание масок;
использование LiDAR в приложении.

Домашние задания
Обучение Классификатора или обжект идентификатора
Цель
Потренироваться в создании моделей для CoreML

Модуль 8. Мультиплатформенная разработка: перенос на Android, Vulkan/Metal

Тема 1
Мультиплатформа для Rich Media: Metal и Vulkan, разработки игр, Video/Image процессинг

Цель занятия
писать кроссплатформенный GPU код;
настраивать окружение для этого.

Краткое содержание
GPU конвейер;
Metal;
OpenGL;
MoltenVK.

Тема 2
Одновременная реализация фич на iOS + Android. Необходимый tool-set

Цель занятия
настраивать окружение;
собирать KMM стек для iOS;
решать проблемы с gradle.

Краткое содержание
как Swift разработчику писать на Kotlin;
SwiftUI + Jetpack Compose;
Combine + RxAndroid/RxKotlin/RxJava.

Домашние задания
Создание мультиплатформенного сетевого слоя с помощью KMM и openapi-generator
Цель
Научится собирать мультиплатформу с кодогенерацией.

Тема 3
Jetpack Compose

Цель занятия
сравнить SwiftUI и Compose;
переносить логику разработки на SwiftUI в Compose.

Краткое содержание
Android Studio & IntelliJ IDEA;
обзор Kotlin Multiplatform и целесообразность общего ядра;
разработка Pure Swift + Pure Kotlin.

Модуль 9. Организация разработки

Тема 1
Тестирование кода XCTest, UITest, fastlane и CI

Цель занятия
собрать CI (Continuous Integration) на fastlane;
использовать XCTest.

Краткое содержание
test-Driven Development (TDD);
теория тестирования;
зачем нужен UITest.

Тема 2
Как правильно написать резюме и развивать hard-skills

Цель занятия
корректно писать резюме;
выбирать работодателя, чтобы развивать свои hard-skills.

Краткое содержание
какие бывают работодатели;
какие скрытые критерии отбора используются.

Домашние задания
Написание работающего резюме
Цель
Создать/доработать резюме, чтобы получать больше откликов и оно меньше вызывало вопросов со стороны рекрутеров и работодателей

Модуль 10. Проектная работа

Тема 1
Написание приложения с нуля

Цель занятия
выбрать и обсудить тему проектной работы;
спланировать работу над проектом;
ознакомиться с регламентом работы над проектом;
генерировать идеи для простых приложений на основе известных «болей» пользователей;
использовать iOS платформу для генерации идей для приложений.

Краткое содержание
подбор инструментов, помощь с стартом написания приложения;
правила работы над проектом и специфика проведения итоговой защиты;
требования к результату проекта и итоговой документации.

Домашние задания
Проектная работа
Цель
Создание проектной работы, которая будет указана в сертификате.

Тема 2
Консультация по проектам и домашним заданиям

Цель занятия
получить ответы на вопросы по проекту, ДЗ и по курсу.

Краткое содержание
вопросы по улучшению и оптимизации работы над проектом;
затруднения при выполнении ДЗ;
вопросы по программе.

Тема 3
Защита проектных работ

Цель занятия
защитить проект и получить рекомендации экспертов.

Краткое содержание
презентация проектов перед комиссией;
вопросы и комментарии по проектам.