

01. Основы программирования

Описание курса

Курс "Основы программирования" - это курс, который познакомит вас с основными терминами и понятиями, используемыми в программировании. Анимационная презентация курса призвана доступно и наглядно продемонстрировать все процессы, происходящие в программировании, для лучшего понимания и осознания.

Вам нужен этот курс:

- если вы совсем не знакомы с программированием;
- если вы не знакомы с языками программирования;
- если вы только начинаете свой путь программиста и не знаете, с чего начать;
- если вы хотите укрепить свою базу и еще раз «разложить по полочкам» уже имеющиеся у вас знания. Повторение во многом, не только мать учения, но и отец понимания!

Для того, чтобы построить хороший дом, нужно заложить надежный фундамент. Этот фундамент - наш курс "Основы программирования".

Содержание курса

01. Для кого этот курс (01:35)
02. Определения (04:44)
03. Язык программирования (04:20)
04. Исходный код (02:57)
05. Компилятор и интерпретатор (04:00)
06. Почему Swift (01:52)
07. Константы и переменные (03:35)
08. Типы данных (03:01)
09. Имена (02:19)
10. Базовые операторы (03:12)
11. Оператор if (02:21)
12. Оператор switch (02:29)
13. Разделение кода (04:24)
14. Функции (02:09)
15. Аргументы и возвращаемые значения функций (04:52)
16. Интерполяция строки. Функция print() (02:03)
17. Комментирование кода (02:13)
18. Область видимости (02:45)
19. Циклы (04:06)
20. Типы коллекций (00:56)
21. Массивы (03:00)
22. Словари (03:30)
23. Множества (01:38)
24. Стиль написания кода (04:07)
25. Объектно-ориентированное программирование (05:09)
26. Ищем и исправляем ошибки (07:06)
27. Многопоточность (02:34)
28. Фреймворки и библиотеки (04:04)
29. Спасибо! (00:58)

Общая продолжительность курса 1:31:59

02. Гид по языку программирования Swift

Описание курса

Язык программирования Swift является новым языком корпорации Apple, который призван заменить собой Objective-C, существующий с середины 80-х годов. В этом курсе вы узнаете все основные моменты языка, что позволит вам без проблем написать свой отлично работающий код. Язык Swift является новейшей разработкой Apple и имеет ряд преимуществ и отличий от языков семейства C, с которыми вы так же познакомитесь и будете работать в этом курсе.

Вам не стоит переживать, что вы чего-то не поймете или что вы не с того языка начинаете свой путь программиста. Совершенно нет. Язык Swift идеален для старта начинающим программистам, так как не имеет запутанного синтаксиса и лишнего багажа, который тащат большинство языков еще с 70-х годов, так как он был представлен только в 2014 году на WWDC (международная конференция разработчиков Apple). С помощью playground (встроенная функциональность Xcode, начиная с версии Xcode 6) вы сможете освоить язык за короткие сроки, так как весь ваш код выполняется моментально и у вас нет необходимости создавать проекты или каждый раз запускать симулятор.

Да, вам придется открыть для себя много нового, но если в вас есть желание изучать, познавать, а затем создавать что-то новое, то это именно для вас!

Содержание курса

01. Знакомство с Playground (11:07)
02. Константы и переменные (04:20)
03. Типы данных (08:26)
04. Базовые операторы (13:46)
05. Интерполяция строк (04:44)
06. Массивы (10:32)
07. Словари (07:32)
08. Множества (08:53)
09. Инструкция if (10:30)
10. Инструкция guard (05:30)
11. Инструкция switch (05:21)
12. Цикл for-in (08:16)
13. Циклы while, repeat-while (03:38)
14. Функции (17:51)
15. Замыкания или клоужеры (05:26)
16. КORTEжи (06:46)
17. Опционалы (06:13)
18. Классы (09:08)
19. Инициализаторы (06:01)
20. Наследование (09:16)
21. Вычисляемые свойства (10:22)
22. Свойства классов (06:09)
23. Ленивые свойства (02:46)
24. Наблюдатели свойства (05:49)
25. Уровни доступа (07:06)
26. Алиасы типов (04:02)
27. Перечисления или энумы (11:05)
28. Структуры (08:02)
29. Импорт фреймворков или библиотек (05:53)
30. Проверка типов и приведение типов (10:43)
31. Типы AnyObject и Any (03:56)
32. ARC (09:10)
33. ARC в замыканиях (09:04)
34. Опциональные цепочки (06:14)
35. Обработка ошибок и отложенные действия (15:22)
36. Сабскрипты (06:15)
37. Расширения (04:48)
38. Протоколы (10:54)
39. Универсальные шаблоны (09:15)
40. Обновление Swift 4 (08:14)
41. Новые фишки Swift 5 (05:44)
42. Заключение (00:57)

Общая продолжительность курса 5:25:06

03. Мои первые приложения

Описание курса

Здорово, когда уже начинаешь разбираться понемногу в языке программирования и еще лучше, когда ты можешь что-то при помощи него сделать. Процесс обучения программирования затягивает и всегда хочется двигаться к новым и новым горизонтам. Этот курс является для нас с вами первой вешкой, которая будет уже означать, что вы разработчик, так как тут мы с вами создадим наши с вами первые приложения!

Первые приложения всегда связаны с некоторой новой информацией, без которой ну совсем никак не создать ни одного приложения, так как помимо кода нам с вами предстоит поработать и с другими моментами "производства" приложения.

Мы с вами не будем затрагивать те темы, которые могут сразу вам не понадобиться, а сосредоточимся на том, что действительно важно для первого скорого результата. А именно мы поговорим о самом Xcode, поработаем хорошенько с интерфейс билдером, так как интерфейс приложения - лицо приложения и нам с вами нужно уметь его создавать, а также мы поработаем с кодом, отчасти с которым вы уже знакомы.

По завершению данного курса у вас будет хорошая практика по работе с интерфейс билдером, понимание работы с кодом и приложением в целом внутри Xcode, а также необходимый фундамент и, конечно, мотивация для вашего быстрого движения вперед!

Содержание курса

01. Intro (02:32)
02. Recap basics (19:05)
03. Basics of Xcode (07:52)
04. First project review (14:59)
05. Interface builder basics (11:39)
06. Interface builder basics 2 (16:46)
07. Interface Builder basics 3 (07:40)
08. Interface builder Basics. Constraint Priority (09:46)
09. IBOutlets, IBActions (12:48)
10. Common mistakes (05:09)
11. Weekday Finder. Interface (08:16)
12. Weekday Finder. Finished with interface builder (11:31)
13. Weekday Finder. Working with code (19:21)
14. TConverter interface elements (06:24)
15. TConverter. Fixing interface elements (09:28)
16. TConverter. Working with code (12:53)
17. PassDataProject. Interface (15:44)
18. PassDataProject. Segues (10:37)
19. PassDataProject. Unwind segue (10:48)
20. Outro (00:54)

Общая продолжительность курса 3:34:12

04. Фреймворк UIKit

Описание курса

Вы продолжаете изучать программирование на Swift и это просто классно. Этот курс так же нацелен на начинающих, которые еще делают свои первые шаги в программировании и еще не знакомы с основами работ с пользовательским интерфейсом.

Вы наверняка обратили внимание на то, что курс называется "фреймворк UIKit", а это значит, что здесь мы с вами познакомимся с фреймворком UIKit, который отвечает за все элементы пользовательского интерфейса (т.е. кнопки, картинки, слайдеры, текстовые поля, жесты и т.д.), что можно понять по приставке UI - User Interface (пользовательский интерфейс). В этом курсе мы с вами познакомимся со всеми необходимыми элементами пользовательского интерфейса, чтобы вы могли написать любое приложение и не чувствовали неловкость при взгляде на библиотеку объектов.

Здесь мы не просто познакомимся с элементами UI, но и разберем работу с каждым из них, так что вы сможете применять свои знания на практике сразу после просмотра конкретного видео. Чтобы вам было комфортно проходить данный курс, убедитесь в том, что вы знакомы с основами языка Swift, так как здесь мы с вами на основах самого языка подробно останавливаться не будем, а будем исключительно использовать наши знания языка для практики с элементами UI. Ну что, ребята, к новым целям и новым достижениям!

Содержание курса

01. Intro (04:53)
02. UIButton и UILabel (22:50)
03. UISegmentedControl (09:55)
04. UISlider (13:01)
05. UITextField (18:34)
06. UIScrollView (06:34)
07. UIDatePicker (11:38)
08. UISwitch (07:16)
09. UIPickerView (19:46)
10. UITextView (23:08)
11. UIStepper (05:26)
12. UIActivityIndicatorView (06:11)
13. UIProgressView (04:29)
14. UITableView UIImage (16:39)
15. UICollectionView (09:52)
16. UIStackView (17:50)
17. UITabBar (05:13)
18. WebKit (15:19)
19. UIContainerView (15:01)
20. UIPageViewController. Part 1 (10:18)
21. UIPageViewController. Part 2 (13:24)

22. UINavigationController. Part 3 (09:02)
23. UINavigationController. Part 4 (05:38)
24. UIGestureRecognizer (11:36)
25. Outro (01:04)

Общая продолжительность курса 4:44:37

05. Приложение на основе UITableView

Описание курса

Если вы знакомы с языком Swift и решили создавать ваши приложения, то вам обязательно пригодится этот курс! Не имеет значения на основе чего вы делаете приложения, является важным общее знание принципа работы с программой Xcode. В этом курсе мы вместе с вами создадим приложение, которое познакомит вас с фреймворками UIKit, CoreData, MapKit, WebKit, CloudKit.

Мы разберем просто тонну нового и интересного прямо на практике. Вы своими руками создадите приложение, которое будет иметь несколько экранов, между которыми вы будете передавать различные данные. Вы научитесь размещать элементы и закреплять их, используя простые constraints или относительно новый stack view. Тут мы с вами посмотрим, как можно хранить данные в базе данных и реализуем алгоритм сохранения, загрузки данных из самой базы данных, используя CoreData. И более того, мы с вами напишем код, который позволит нам хранить данные в iCloud, а также код, который позволит нам получать их оттуда. Мы так же поработаем с картами, с поиском, с браузером и многое, многое другое.

Вместе с вами мы создадим все с самого нуля и разберем все сами и на своем опыте.

Курс написан на Swift 4.0, что позволит ваш шагать в ногу со временем!

Пусть вас не смущает меню на английском языке. Мы хотим, чтобы вы, проходя наши курсы, постоянно слышали новые термины на английском языке и осваивали их, ну а меню нам в этом немного помогает.

Содержание курса

01. Hello, Xcode! (10:43)
02. First project (10:44)
03. How it works (03:00)
04. Autolayout, constraints (10:51)
05. UIStackView - Part 1 (08:26)
06. UIStackView - Part 2 (04:54)
07. Table view - Part 1 (13:25)
08. Table view - Part 2 (08:43)
09. Main project with UITableViewController (08:25)
10. Custom table view cell (10:43)
11. Custom cell appearance (06:54)
12. UIAlertController (07:04)
13. Cell accessory type (06:16)
14. Cell accessory bug fix (06:01)
15. Default row actions (04:41)
16. Custom row actions (05:26)
17. UINavigationController and Segues (08:09)
18. Passing data between scenes (04:17)
19. Object oriented programming (05:31)
20. Detail view interface - Part 1 (07:09)
21. Detail view interface - Part 2 (04:24)
22. Detail view code update (04:17)
23. Navigation and status bar appearance (09:43)
24. Hide navigation bar on swipe (03:20)
25. Self-sizing cells (05:38)
26. Self-sizing cells for main screen (02:48)
27. Rate screen and unwind segue (07:34)
28. Blurring background (06:14)
29. Animation (08:36)
30. Passing data via unwind segue (06:01)
31. MapKit and Map view (05:31)
32. From address to coordinates (08:11)
33. Map annotations (05:21)
34. New eatery screen (04:03)
35. Interface of new eatery screen (10:44)
36. Class UIImagePickerController (09:28)
37. UIImagePickerControllerDelegate (03:48)

38. Is our already eatery visited (08:20)
39. Introducing Core Data (10:37)
40. Save data using Core Data (05:58)
41. Fetching saved data (06:35)
42. Updating main screen table view with new eatery (06:51)
43. Delete data from data base (02:39)
44. Search feature - Part 1 (08:01)
45. Search feature - Part 2 (06:59)
46. Search feature - Part 3 (06:40)
47. UINavigationController - Part 1 (14:44)
48. UINavigationController - Part 2 (08:42)
49. UINavigationController - Part 3 (04:32)
50. UINavigationController - Part 4 (08:38)
51. UserDefaults class (03:50)
52. UITabBarController class (07:49)
53. About us screen (06:59)
54. Web view - Part 1 (05:14)
55. Web view - Part 2 (10:33)
56. Setting iCloud feature (03:58)
57. CloudKit dashboard (06:38)
58. Retrieving data from iCloud (11:36)
59. Retrieving data with CKQueryOperation class (08:12)
60. Fetching images with CKFetchResultsOperation class (06:38)
61. Adding activity indicator (07:25)
62. Caching images with NSCache (04:59)
63. Pull-to-refresh feature (04:38)
64. Save data to iCloud (11:17)
65. Localization (09:57)
66. Icons and launch screen (02:59)
67. Outro (00:45)

Общая продолжительность курса 7:49:46

05. Приложение на основе UITableView (Swift 5)

Описание курса

Если вы уже знакомы с основами языка Swift и решили попробовать свои силы в создании полноценного приложения, то вам обязательно пригодится данный курс! На этом курсе мы с вами вместе создадим приложение на основе Table View, в котором будем использовать такие фреймворки, как UIKit, MapKit, а также Realm.

Мы разберем кучу различных концепций прямо на практике. Вы своими руками создадите приложение, которое будет иметь несколько экранов, между которыми будут передаваться различные данные. Сами данные мы будем хранить в базе данных и реализуем алгоритм сохранения, редактирования и загрузки данных, используя Realm. Мы так же поработаем с картами, с поиском, с сортировкой данных, реализуем возможность выставления рейтинга и сделаем еще многое и многое другое.

Курс написан на Swift 5.0, что позволит вам шагать в ногу со временем!

Пусть вас не смущает меню на английском языке. Мы хотим, чтобы вы, проходя наши курсы, постоянно слышали новые термины именно на английском и осваивали их, ну а меню вам в этом немного поможет.

Рекомендуется к изучению после прохождения следующих курсов:

- Основы программирования
- Гид по языку программирования Swift
- Фреймворк UIKit
- Мои первые приложения

Данный курс предназначен для начинающих разработчиков, которые уже знакомы с основами языка Swift, с его синтаксисом и с его основными концепциями. Предполагается, что вы уже знакомы со всеми основными инструментами для работы со Storyboard, а также понимаете концепцию Auto Layout.

Содержание курса

01. Intro (05:08)
02. Create Table View App (15:52)
03. Documentation (10:01)
04. Configure the cell (13:20)
05. Work with images (07:45)
06. Design the App (14:41)

07. Custom Table View Cell (13:46)
08. Work with Model (11:55)
09. New Place ViewController (18:06)
10. Design for NewPlaceViewController (17:39)
11. Image Picker Controller (10:32)
12. Image Selection (06:54)
13. Action Sheet Customization (04:07)
14. Pass Data between Controllers (23:04)
15. Introduction to the Realm (08:01)
16. Model Setup (15:27)
17. Refactor Code to Use DB (11:27)
18. Delete Action (04:43)
19. Edit Action (18:07)
20. Prepare for Sorting (09:01)
21. Sorting (13:07)
22. UISearchBar (18:27)
23. Custom Rating Control - Part 1 (23:32)
24. Custom Rating Control - Part 2 (18:24)
25. Save Rating to Database (07:12)
26. Rating with Cosmos Star (22:26)
27. Map View Controller (07:46)
28. Setup Placemark (10:25)
29. Work with Annotations (09:56)
30. Bug Fixes (05:59)
31. User Location (14:02)
32. Center View In User Location (08:50)
33. Prepare Map VC to Get Address (16:49)
34. Get Address from Map (09:18)
35. Passed Address to NewPlaceVC (08:11)
36. Get Directions (18:03)
37. Tracking User Location (12:38)
38. Refactoring (18:50)
39. Memory Management Basics (11:01)
40. Finding and Fixing Memory Leaks (14:00)
41. Weak, Strong, Unowned References (11:49)
42. Retain Cycles in Closures (15:16)

Общая продолжительность курса 8:55:37

06. Встроенные покупки (in-app purchases)

Описание курса

Каждый из нас скачивал приложения в AppStore, и с удивлением замечал, что большинство этих приложений бесплатные. Но как же так? Как тогда разработчики зарабатывают себе на хлеб? Тут все просто, в них имеются встроенные покупки. Процесс заключается в следующем: вы платите деньги через приложение, которое соединяется с AppStore и отслеживает вашу транзакцию.

После этого, в зависимости от покупки, на счет разработчика переходит определенная сумма и все, ваш внутриигровой товар оплачен. Красота, да и только! И вариантов встроенных покупок на самом деле не мало, поэтому в этом деле все зависит от жадн...кхм-кхм фантазии разработчика. Так давайте же познакомимся с ними поближе!

Для этого курса вам понадобится Xcode 9 и лицензия разработчика Apple, так как тестировать вам придется ваши покупки только на реальном устройстве на iOS 11 .

Можно представить, как большинству разработчиков скорее хочется окупить свои усилия разработки приложения и поскорее познакомиться со встроенными покупками или IAP (in-app purchase). Если вы пока что общаетесь со Swift на "Вы", то вам лучше потратить денек-другой времени на повторение теоретической базы Swift, чтобы полностью погрузиться в процесс встроенных покупок и не тратить время на повторение букваря :). Ну все, не хочется отнимать ваше драгоценное время, погрузимся в мир встроенных покупок!

Поехали!

Содержание курса

01. Интро (01:48)
02. Теория (08:51)
03. Настройка itunesconnect (04:28)
04. Создаем встроенные покупки (13:15)
05. Обзор стартового проекта (04:40)
06. Настраиваем покупки (11:18)

07. Получаем товары (06:03)
08. Экран покупок (10:13)
09. Совершаем сделку (12:06)
10. Совершаем сделку (07:44)
11. Определяем совершенную покупку (09:58)
12. Подготовка проекта к валидации чека (07:23)
13. Валидация чека (13:42)
14. Спасибо! (00:30)

Общая продолжительность курса 1:51:59

07. Изучаем основы Core Data

Описание курса

Рано или поздно каждый из нас сталкивается с необходимостью сохранения данных на устройстве. Вот настал и ваш черед узнать, как можно это сделать при помощи нативного фреймворка CoreData.

В этом курсе мы с вами познакомимся с основами работы фреймворка CoreData и при помощи полученных знаний создадим три приложения, чтобы их закрепить.

Вы решили стать разработчиком? Не сомневайтесь, этот курс вам обязательно пригодится! Поехали!

Содержание курса

01. Intro (01:17)
02. What is CoreData (04:27)
03. To-do list without CoreData (08:22)
04. Save data using CoreData (09:23)
05. Finishing to-do list app (02:19)
06. Second project (03:03)
07. Creating data model (03:11)
08. Two moments (05:54)
09. Get data from .plist file (16:04)
10. Update interface with data (08:30)
11. Start and Rate buttons functionality (09:10)
12. Segmented control. Project review (08:25)
13. Third project. Core Data stack (04:34)
14. Core Data stack (06:31)
15. Data model (06:42)
16. Save and fetch data from Core Data (08:58)
17. Delete data. Project review (08:31)

Общая продолжительность курса 1:55:21

08. Изучаем Grand Central Dispatch

Описание курса

Курс по GCD (Grand Central Dispatch) предназначен для того, чтобы вы освоили теорию многопоточного программирования и могли применить ее на практике. Здесь мы с вами познакомимся с основными понятиями GCD, с основными методами GCD, которые позволят вашему приложению работать, используя сразу несколько потоков, что может значительно улучшить производительность работы ваших приложений.

При прохождении этого курса желательно, чтобы вы знали основы языка Swift и имели базовое представление о программе Xcode и создании приложений.

Содержание курса

01. Вступление (00:50)
02. Потоки (05:20)
03. Очереди (12:12)
04. Синхронность и асинхронность (01:40)
05. Последовательность работы с очередями (02:44)
06. Интерфейс приложения (05:16)
07. Проблема приложения (10:46)
08. Применяем GCD (05:24)
09. Задержка asyncAfter (10:15)
10. Новая очередь. Барьеры отправки (07:46)
11. Практикуемся с барьерами (11:35)
12. Группы отправки (05:47)
13. Группы отправки. Практика (09:28)
14. Блоки отправки (02:54)

15. Блоки отправки. Практика (08:00)
16. Семафоры отправки (04:38)
17. Семафоры отправки. Практика (06:33)
18. Источники отправки (03:41)
19. Источники отправки. Практика (04:22)
20. Спасибо! (00:45)

Общая продолжительность курса 1:59:56

09. Application Programming Interface

Описание курса

Большинство приложений AppStore так или иначе работают с различными API. А API - это интерфейс взаимодействия вашего приложения и какого-либо ресурса, то есть, другими словами, вы можете брать информацию с какого-либо ресурса и размещать ее у себя в приложении именно в том виде, который нужен вам или вашему пользователю.

А осваивать новую для нас тему мы будем с вами на примере приложения, отображающего погоду того места, где вы сейчас находитесь. Так же параллельно с изучением основной темы мы с вами активно потренируемся использовать классы, структуры, перечисления, протоколы, расширения протоколов, универсальные шаблоны, Sprite Kit Scene, Stack View и прочее. Лучшее всего этот курс подойдет для тех, кто знаком с основами языка Swift. Поэтому если при прохождении этого курса, чувствуете, что ничего не понимаете, то возможно вам стоит вернуться и пройти базу.

Курс записан на русском языке, но уделяем внимание так же и английским терминам, чтобы у вас росло восприятие английского.

Содержание курса

01. Intro (01:11)
02. Add elements to our interface (06:20)
03. UIStackView (09:50)
04. Available forecast data (05:26)
05. Weather Icons (08:24)
06. Fake data try (10:59)
07. API & ReST API (05:46)
08. HTTP Request (04:15)
09. Unsafe Networking Code (07:54)
10. APIManager (06:40)
11. APIManager Completion (04:33)
12. Protocol extension (06:02)
13. Default Implementation - Part 1 (03:25)
14. Default Implementation - Part 2 (04:38)
15. API Weather Manager (04:41)
16. Composite URL (07:06)
17. Failable initialization (08:02)
18. Getting real data (07:25)
19. Activity Indicator (04:20)
20. App Transport Security Settings (02:02)
21. Sprite Kit Effects (05:21)
22. Particles Settings (03:38)
23. Apply Rain Effect (05:23)
24. Retrieving coordinates (06:19)
25. Outro (00:48)

Общая продолжительность курса 2:20:28

10. Изучаем класс и подклассы CALayer

Описание курса

Вы уже разбираетесь в Swift и хотите получить новую информацию, которая актуальна для любого проекта? Тогда вы по адресу!

Этот новый курс посвящен целиком и полностью классам CALayer и его подклассам: CAShapeLayer, CAGradientLayer, CAReplicatorLayer, CABasicAnimation. Префикс CA обозначает, что мы будем работать с классами фреймворка Core Animation, который в свою очередь необходим нам для работы с любым элементом, который либо наследуется, либо является экземпляром UIView. Так же мы с вами поработаем с классом UIBezierPath, который пригодится нам для создания маски для создания нашей собственной, кастомной формы элемента.

Этот курс предназначен для тех, кто уже имеет хотя бы небольшой опыт в программировании на Swift, но очень хочет исследовать новые горизонты и не боится экспериментов!

Пусть вас не смущает меню на английском языке. Мы хотим, чтобы вы, проходя наши курсы, постоянно слышали новые термины на английском языке и осваивали их, ну а меню нам в этом немного помогает.

Содержание курса

01. Intro (00:52)
02. CALayer (07:28)
03. Demo project (11:19)
04. CAGradientLayer (13:42)
05. CAShapeLayer (13:31)
06. CABasicAnimation (05:52)
07. Mask. @IBDesignable and @IBInspectable (12:58)
08. CAReplicatorLayer (14:03)
09. Outro (00:44)

Общая продолжительность курса 1:20:29

11. Изучаем систему Git. GitHub

Описание курса

Вы уже разбираетесь в Swift выросли до разработки в команде? Вы готовы создавать ваши проекты правильно с использованием ветвления через git, чтобы можно было их легко поддерживать и развивать? Ну тогда вы в правильном месте!

В этом курсе мы с вами подробно поговорим о разработке в команде с использованием git и GitHub. Мы подробно рассмотрим такие понятия как: repository, commit, branch и многое-многое другое, начиная с самых теоретических основ и заканчивая практикой.

Этот курс имеет крайне широкое предназначение, потому что распространяется не только для разработчиков под iOS, но и для разработчиков вообще любого профиля. Так что если вы попали сюда как разработчик программ не предназначенных для iOS, то вы все равно по адресу.

Пусть вас не смущает меню на английском языке. Мы хотим, чтобы вы, проходя наши курсы, постоянно слышали новые термины на английском языке и осваивали их, ну а меню нам в этом немного помогает.

Содержание курса

01. Intro (01:12)
02. Git (07:16)
03. Commits and repositories (05:37)
04. Git practice (10:43)
05. Branch (13:12)
06. Remote repository (04:03)
07. Remote repository practice (12:47)
08. Merge conflict (09:42)
09. Desktop GitHub (14:49)
10. Useful resources (06:04)
11. Outro (01:00)

Общая продолжительность курса 1:26:25

12. Firebase - Наше первое приложение

Описание курса

Новый курс "Firebase: Наше первое приложение" познакомит Вас с основами сервиса Firebase, где мы с вами познакомимся с двумя наиболее востребованными сервисами Firebase при создании практически любого приложения - это Firebase Authentication (аутентификация пользователя) и Firebase Realtime Database (база данных, которая моментально обновляется и позволяет отображать изменения сразу после того, как они наступили).

Так как этот курс сосредоточен на работе с Firebase, то я подразумеваю, что вы уже неплохо знакомы с основами самого языка Swift. Если вдруг это не так, то сначала возможно Вам лучше пройти наш первый курс по основам Swift.

Если рассказать подробнее, то в этом курсе мы с вами создадим приложение на подобие To-do List (список задач/список покупок), где каждый пользователь обязательно должен зарегистрироваться, после чего уже сможет добавлять себе задачи. Данные регистрации пользователя и сам список задач пользователя будут храниться в самом сервисе Firebase, и именно на примере работы с этими данными мы и познакомимся с Firebase. И, конечно, все приложение мы с вами сделаем с самого нуля!

И, как всегда, хочу отметить, что пусть Вас не смущает меню на английском языке. Мы ведь с вами так же немного подтягиваем свой английский, верно?

Содержание курса

01. Intro (01:17)
02. Creating new project (12:50)
03. Creating tasks screen (16:45)
04. Firebase (06:21)
05. Keyboard issue (09:44)
06. Authentication (20:05)
07. Task form. JSON Data (17:19)
08. Save data (14:17)
09. Fetch data (06:59)
10. Remove data (07:17)

Общая продолжительность курса 1:52:54

13. Фреймворк для создания игр SpriteKit

Описание курса

Представляем вам долгожданный курс по SpriteKit - курсу по созданию игр!

Над этим курсом мы усердно работали, чтобы вы смогли не просто написать игру, но и познакомиться с важными аспектами программирования, которые используются не только при создании игр, но и в программировании на Swift в целом.

В этом курсе мы:

- подробно разберем темы создания классов, подклассов с использованием простого наследования, либо через статические методы
- рассмотрим положительные и отрицательные стороны каждого из подходов на практике, что позволит вам крепко-накрепко закрепить фундамент ваших знаний
- будем работать с синглтоном, который будет хранить нашу игровую сцену, пока мы прохлаждаемся на паузе
- создадим игру с четким разделением всего проекта на логические части, так что вам будет намного понятнее на какие куски вам нужно будет разбивать ваши будущие проекты
- рассмотрим работу с протоколами и расширениями, которые позволяют расширить функционал ваших классов (да и структур тоже) фактически не увеличивая их.

При написании игры мы с вами познакомимся:

- с анимацией персонажа
- с действиями
- с emitter node, который отрисовывает анимацию частиц по нашим настройкам
- со сценами
- переходами между сценами
- с масками и методами создания этих масок
- с физическими телами и многим-многим другим...

Мы с вами напишем полноценную игру, которую вы сможете выложить в AppStore!

Пусть вас не смущает меню на английском языке. Мы хотим, чтобы вы, проходя наши курсы, постоянно слышали новые термины на английском языке и осваивали их, ну а меню нам в этом немного помогает.

Единственным условием для прохождения курса - наличие iPhone для тестирования вашей игры, так как на симуляторе работать она не будет!

Содержание курса

01. Intro (00:56)
02. Demo project (23:56)
03. Cleaning project (06:15)
04. Adding background (10:22)
05. Adding islands (16:35)
06. Adding clouds (13:55)
07. Player plane class (08:03)
08. Movement by CoreMotion (15:20)
09. Islands movement (14:16)
10. Moving background (07:00)
11. Remove from scene (08:53)
12. Cleaning our project (07:40)
13. Turning plane animation (12:55)
14. Turning plane animation. Part 2 (23:31)
15. Power Up (11:58)
16. Little code refactoring (13:35)
17. Creating class Enemy (09:06)
18. Enemy movement (15:14)
19. Enemy final movement (19:19)
20. PowerUp classes (14:15)

21. PowerUp movement logic (11:26)
22. Player fire ability (11:11)
23. Preload atlases (10:13)
24. Menu scene (09:07)
25. Physics body (18:47)
26. Collisions (07:36)
27. Physics body with CGPath (06:40)
28. Bit masks advanced (14:33)
29. User Interface (21:48)
30. Make it nice (06:39)
31. Menu interface (19:17)
32. Pause scene (05:46)
33. Scene manager (10:54)
34. Pause logic (12:33)
35. Scene logic (18:50)
36. Best scene (10:39)
37. Collision explosion (11:07)
38. Update lives (07:51)
39. Game over scene (09:27)
40. Home Task Wrap Up (07:31)
41. Music and sound (06:14)
42. UserDefaults (14:52)
43. Scores (08:20)
44. Publish your awesome app (11:21)
45. Outro (00:42)

Общая продолжительность курса 8:46:28

14. Sketch - Рисуем интерфейсы

Описание курса

Это наш первый курс по работе с дизайном и это первый курс по работе с такой классной, перспективной и одновременно простой программой для отрисовки невероятных логотипов и интерфейсов - Sketch.

Так как Sketch является приложением, которое невероятно хорошо заточено для отрисовки интерфейсов, то просмотрев этот курс вы сможете абсолютно самостоятельно рисовать интерфейсы для ваших приложений, не тратя деньги на дорогостоящие наборы. Вы сможете создавать именно то, что нравится вам!

Курс подходит:

- идеально для начинающих
- для тех, у кого нет опыта работы с графикой
- очень хорош для разработчиков
- для обучения работы с любыми интерфейсами
- для изучения инструментов работы с графикой
- для обучения работы с вектором
- для получения полного цикла знаний для самостоятельной работы с графикой
- для получения даст полный цикл знаний для разработки интерфейсов любой сложности

На момент записи курса было выпущено небольшое обновление Sketch до версии 44.1. Курс записан на версии Sketch 43.1 и является полностью совместимым с более новой версией 44.1.

Содержание курса

01. Intro (02:05)
02. About Sketch Application (07:06)
03. First steps in Sketch (15:48)
04. Email App Login screen (17:23)
05. Email App Inbox screen (24:10)
06. Email App Menu screen (13:13)
07. Email App Reading mail screen (09:56)
08. Email App New message (07:02)
09. Nice App Water interface (18:21)
10. Nice App Speed Test (20:56)
11. Nice App To-Do List (18:06)
12. Nice App Loading screen (20:26)
13. Nice App Testimonials (13:43)
14. Nice App Export (07:23)
15. Outro (00:34)

Общая продолжительность курса 3:16:12

15. Паттерны проектирования

Описание курса

Программистам нужно знать паттерны. Они очень помогают в работе, облегчают жизнь и просто говорят о том, что вы то уже "не лыком шиты"! Что же такое паттерны? Это отточенные лучшие практики или решения для стандартных задач в объектно-ориентированном программировании. Паттернов достаточно много и более того, зачастую их перемешивают между собой, чтобы получить наиболее рациональные решения для поставленной задачи.

Какую проблему решают паттерны?

Вы пишете программы/приложения и публикуете их. Придет время, когда вам нужно будет обновить ваше приложение или программу и в зависимости от того насколько хорошо написан ваш код, столько сил на обновление вам и придется приложить. То есть если вкратце, то паттерны при правильном использовании могут сэкономить вам кучу времени на поддержке вашего проекта.

Также паттерны решают проблему коммуникации между программистами одной команды, так как зная паттерны, вам не придется вдаваться в подробности собственного варианта решения задачи, потому что другие члены команды будут понимать вас с полуслова.

Чему вы научитесь в этом курсе?

Так как курс построен по принципу "презентация - работа в Xcode - презентация", то в первой половине урока вы будете смотреть презентацию того, как мог бы выглядеть паттерн в нашей реальной жизни, а после этого мы с вами будем перекладывать пример паттерна уже на код на Swift, а в хвостике урока мы с вами будем разбирать паттерн схематично.

Вы научитесь понимать, как разбивается код для работы с теми или иными паттернами и научитесь находить паттерны в чужом коде, что сделает этот код в разы понятнее. Вы так же уже гарантированно будете размышлять перед тем, как бросаться кодить, так как зная паттерны всегда хочется сделать код лучше :). Кстати, именно эта черта отличает программиста от хорошего программиста!

Курс подходит:

- для начинающих разработчиков, кто уже имеет небольшой опыт работы со Swift
- для тех, кто уже умеет работать со Swift, но пока не решался познакомиться с паттернами
- для тех, кто уже программировал на другом объектно-ориентированном языке, но решил перейти в Swift и сразу окунуться в паттерны
- для обучения работы с протоколами
- для изучения более глубокого подхода к коду
- для тех, кто хочет на практике познакомиться с таким понятие как "полиморфизм"
- кто хочет получить не устаревающие знания, которые можно перенести в любой объектно-ориентированный язык программирования
- для хардкодеров, которым нравится много кода, очень много кода!

Содержание курса

01. Intro (08:47)
02. Strategy (14:55)
03. Observer (12:57)
04. Decorator (16:18)
05. Simple Factory (12:44)
06. Factory Method (10:07)
07. Abstract Factory (10:05)
08. Singleton (10:29)
09. Command (24:34)
10. Adapter (09:22)
11. Facade (11:12)
12. Template Method (20:02)
13. Iterator (26:09)
14. Composite (12:18)
15. State (18:08)
16. Proxy (15:49)
17. Builder (14:05)
18. Chain of responsibility (12:52)
19. Outro (00:42)

Общая продолжительность курса 4:21:35

16. Изучаем знаменитый паттерн MVC

Описание курса

Этот видео курс посвящен базовой реализации архитектуры MVC или Model View Controller, которая является рекомендуемой архитектурой для создания приложений компанией Apple. MVC является самой простой архитектурой, которую нужно научиться применять, чтобы ваши контроллеры не выростали до огромных размеров, что впоследствии ведет к проблемам развития приложения, его тестирования и просто к проблемам передачи кода другим разработчикам.

В этом курсе мы:

- разберем на простом примере работу архитектуры MVC
- разберем теоретическую составляющую архитектуры MVC
- посмотрим на реализацию архитектуры MVC с одним дополнительным сервисным сетевым слоем
- с вами пробежимся уже по готовому коду, чтобы убедиться, что мы ничего с вами не пропустили и не упустили.

Насколько бы не был сложным ваш проект, на примере нашего тестового одноэкранного приложения мы сами разберем все необходимое, чтобы вы могли переложить архитектуру MVC на ваш существующий проект или на ваши новые проекты.

У нас есть рекомендация по прохождению данного курса, а именно очень желательно, чтобы вы уже имели представление об основах языка Swift, были знакомы с замыканиями, были знакомы с паттерном Singleton, который у нас будет представлять менеджера по работе с сетью.

Содержание курса

01. Introduction (02:10)
02. Theory (11:33)
03. Project - Part 1 (10:45)
04. Project - Part 2 (11:13)
05. Network Layer (10:11)
06. Comments (10:37)
07. Code review (08:49)
08. Outro (00:30)

Общая продолжительность курса 1:05:48

17. Дополненная реальность. ARKit

Описание курса

Вы скорее всего уже не раз слышали о приложениях, где пользователи могут смотреть через камеру своего смартфона на объекты, которых в реальности нет и вот как раз такие приложения имеют дело с дополненной реальностью. На WWDC-2017 Apple анонсировала свой новый фреймворк - ARKit, который предназначен для создания приложений с дополненной реальностью для мобильных устройств Apple работающих на iOS, то есть iPhone и iPad.

В этом курсе мы с вами познакомимся с дополненной реальностью и заодно познакомимся с фреймворком SceneKit, который послужит нам помощником для создания приложений, где мы сможем размещать виртуальные объекты в нашем реальном мире на совершенно реальных поверхностях.

Для этого курса вам понадобится Xcode 9 , базовые знания языка Swift, а также реальное устройство, работающее на iOS 11:

- iPhone SE
- iPhone 6s
- iPhone 6s Plus
- iPhone 7
- iPhone 7 Plus
- 9.7-inch iPad (2017 model)
- 10.5-inch iPad Pro
- 12.9-inch iPad Pro (2017 model)
- 9.7-inch iPad Pro (2016 model)
- 12.9-inch iPad Pro (2016 model)
- или более новое устройство.

Содержание курса

01. Intro (02:51)
02. Theory of ARKit (16:26)
03. Default project review (08:28)
04. First custom AR object on the scene (08:56)
05. Text with cube (05:58)
06. Textures and figures (12:55)
07. Tap Gesture (10:36)
08. Debug options (17:27)
09. Plane class (12:31)
10. Planes (10:27)

11. Placing objects on planes (13:22)
12. Physics bodies (06:57)
13. Collision detection (14:47)
14. 3D Model file formats (09:28)
15. SCNReferenceNode (11:34)
16. Placing 3D object (08:15)
17. Preparation for particles (12:12)
18. Smoke particles (08:01)
19. House with smoking chimney (08:29)
20. Shadows (13:52)
21. Outro (00:44)

Общая продолжительность курса 3:34:16

18. Изучаем паттерн MVVM

Описание курса

Вы уже наверняка знакомы с паттерном (или архитектурой) приложения MVC. И наверняка вы уже успели столкнуться с тем, что ViewController в этом паттерне принимает слишком много обязанностей и начинает "разбухать", так как на него взваливается слишком много работы. Что ж, в этом курсе мы с вами разберем "по косточкам" паттерн MVVM, который пришла на помощь "разбухающему" MVC.

В этом курсе мы с вами подробно разберем паттерн MVVM, посмотрим из чего он состоит, а также посмотрим, как применять его в наших проектах.

Если вы уже знакомы с основами Swift, знакомы с MVC, то вам определенно нужно взглянуть в сторону паттерна MVVM, кто знает, может быть уже сегодня вы начнете внедрять его в свои проекты!

Увидимся в курсе!

Содержание курса

01. Intro (01:27)
02. Theory of MVVM (10:45)
03. MVC project first (06:22)
04. From MVC to MVVM (06:56)
05. UITableView with MVC (11:48)
06. MVVM for UITableViewController. Part 1 (06:38)
07. MVVM for UITableViewController. Part 2 (12:27)
08. MVVM for UITableViewController. Checkpoint (04:30)
09. MVVM and DetailViewModel (02:12)
10. DetailViewController. Part 1 (07:16)
11. DetailViewController. Part 2 (06:34)
12. MVVM for DetailViewController (06:13)
13. Data binding via Boxing (14:30)
14. Working with managers in MVVM (14:36)
15. Outro (00:41)

Общая продолжительность курса 1:52:55

19. Уведомления

Описание курса

Представляем вашему вниманию курс, целиком и полностью посвященный уведомлениям.

Но что такое уведомления? Уведомления - это небольшие сообщения, несущие пользователю важную для него информацию по вашему приложению или по какому-то событию. То есть вы уже догадываетесь, что уведомления играют важную роль во взаимодействии пользователя с приложением и могут вести к увеличению числа контактов пользователя с вашим приложением и, соответственно, к лояльности. Учитывая все это было бы неплохо знать как создавать эти самые уведомления и как кастомизировать их, чтобы точно вписать их в ваши приложения.

В рамках этого курса мы поговорим с вами, как о локальных, так и о пуш-уведомлениях. Вы научитесь создавать меню различных действий для уведомлений, прикреплять контент к уведомлениям и кастомизировать их. Так же мы создадим все необходимые сертификаты и выполним все необходимые настройки для отправки пуш-уведомлений. Поработаем с Firebase и рассмотрим новые фишки уведомлений, которые появились в iOS 12. В общем мы рассмотрим все возможные вариации уведомлений, чтобы у вас была возможность делать не то, что умеете, а то, что нужно!

В общем вас ждет насыщенный, интересный и полезный курс, пройдя который, вы сможете реализовать отправку различных типов уведомлений в своих реальных проектах.

Содержание курса

01. Intro (00:58)
02. Request Authorization (07:00)
03. Local notifications (10:35)
04. Notification handling (05:03)
05. Local Notification with Actions (08:25)
06. Refactoring (04:31)
07. Local Notification with Content (03:31)
08. Intro in Push Notifications (03:40)
09. AppID, SSL Certificate (09:06)
10. UDID and Provisioning Profile (06:49)
11. Register for Push-Notifications (05:04)
12. Sandbox (12:01)
13. Push-Notification with Content (12:49)
14. New Firebase Project (10:35)
15. Push-Notification with Firebase (10:18)
16. NotificationContentExtension. Part 1 (12:19)
17. NotificationContentExtension. Part 2 (17:39)
18. Notifications Settings, Provisional Notifications. Part 1 (07:53)
19. New in iOS 12. Part 2 (07:47)

Общая продолжительность курса 2:36:03

20. Networking

Описание курса

Работая с любыми внешними данными будь то обновление данных о погоде, стриминг музыки или просто серфинг в браузере, ваше приложение использует сетевые запросы HTTP. Именно они отвечают за все, что связано с внешним миром.

Для работы с сетевыми запросам Apple предлагает современный и простой в использовании API URLSession, который представляет из себя объект, координирующий группу связанных сетевых задач для передачи данных. URLSession является ключевым компонентом всего стека, отвечающим за отправку и прием HTTP запросов.

На данном курсе мы подробно разберем стек URLSession, рассмотрим его основные классы и методы и научимся создавать сетевые сессии с различными задачами. Мы пройдем весь путь от создания самого простого GET запроса и до более сложных вариаций POST запросов.

Вы научитесь работать с изображениями в сети, получать данные в формате JSON и декодировать их в свою структуру данных.

Мы поработаем с загрузкой больших объемов информации в фоновом режиме, а также с выгрузкой различных данных на удаленные серверы. Отообразим процесс загрузки данных в интерфейсе приложения и рассмотрим различные возможности параметров конфигурации сетевых запросов.

Так же мы изучим самую популярную стороннюю библиотеку для работы с сетевым кодом Alamofire, которая является оберткой для URLSession и используется для упрощения рутинных действий, связанных с сетевыми запросами.

Ну и в качестве вишенки на торте мы поработаем с авторизацией пользователей в приложении с использованием таких популярных сервисов, как Facebook и Google. Мы поработаем с SDK этих сервисов, настроим для каждого сервиса отдельные кнопки логина, поработаем над кастомизацией этих кнопок, а также с публичными данными пользователей. Для этого мы воспользуемся сервисом Firebase, в котором будем регистрировать всех авторизованных в нашем приложении пользователей и хранить их публичные данные для дальнейшего использования этих данных в приложении.

Содержание курса

01. Intro (01:50)
02. Определение URLSession (04:11)
03. Загрузка изображения из сети (11:02)
04. Терминология (06:31)
05. GET Requests (10:12)
06. POST Requests (09:22)
07. JSONDecoder (19:31)
08. Get and parse data (17:04)
09. Refactoring (30:27)
10. Upload Image (13:06)
11. Background Task Part 1 (17:59)
12. Background Task Part 2 (31:26)
13. URLSessionConfiguration (03:52)
14. Network Request With Alamofire (14:34)

15. Настройка запроса и обработка ответа (06:08)
16. Обработка результата ответа (14:06)
17. Типы ответов (14:03)
18. Download Progress (15:52)
19. POST Request with Alamofire (17:27)
20. BugFixes (04:08)
21. Upload Image with Alamofire (11:19)
22. Обзор Paw (10:06)
23. Facebook SDK (21:53)
24. User Authorization with Facebook (14:51)
25. Custom Facebook Login Button (14:17)
26. Firebase setup (17:06)
27. Fetching Data From FB Profile (05:48)
28. Save Profile Data Into Firebase (17:14)
29. Fetching data from Firebase Database (14:14)
30. Google Sign In (18:13)
31. Log Out (19:21)
32. Custom Google Login Button (05:38)
33. Sign In, Sign Up Introduction (12:58)
34. Activity Indicator on Continue Button (06:16)
35. Sign Up with Email (11:58)
36. Log Out and Fetching Data (06:31)
37. Sign In with Email (05:23)

Общая продолжительность курса 7:55:57

21. TDD. Unit Testing

Описание курса

Работая над любым приложением, вы так или иначе сталкиваетесь с багами, которые нужно править. Но представьте себе, что баги, которые вам встречаются на пути становятся очень редкими, практически как растения из "красной книги". Станет ли разработка привлекательной? Безусловно!

В этом курсе мы с вами познакомимся с Unit тестами и даже более того, мы с вами разберем разработку на основе TDD (Test Driven Development), что в переводе означает разработка через написание тестов. Когда мы используем технику TDD, то сначала мы пишем тест, а только затем пишем код, который должен пройти данный тест. Это отличается от обычного течения событий, так как обычно мы что-то делаем, а потом проверяем, что это что-то работает.

Выше мы с вами несколько раз упомянули слово "тест", но что значит "тест" в данном контексте? В нашем случае под словом тест мы подразумеваем Unit-тест, то тест, который направлен на тестирование одного юнита (блока/куска/фрагмента) кода. По сути, это должен быть небольшой изолированный тест, который просто проверяет, что написанный нами код, действительно делает то, что мы от него ждем.

Наверняка у вас возник вопрос нужно ли проходить именно вам данный курс. Если вы уже прошли основы языка и знаете, как создаются простые приложения, то однозначно этот курс проходить вам нужно, так как тесты очень дисциплинируют ваш код и делают его более качественным. Так же зная как пишутся тесты, зная как разрабатываются приложения по TDD, вы точно будете иметь значительное преимущество в большинстве компаний при трудоустройстве, так как работодатели любят, чтобы их приложения были написаны по TDD. Подробнее о самих тестах, о TDD вы узнаете уже внутри курса, где мы с вами разберем все необходимое, чтобы вы могли самостоятельно использовать Unit-тесты и технику TDD при разработке ваших собственных приложений.

Содержание курса

01. TDD. Unit Testing. Intro (01:24)
02. TDD. Unit Testing. UI Testing (17:43)
03. Demo. Unit Testing (20:28)
04. Preparation for development (05:17)
05. Task development with TDD (09:05)
06. Task Location Property (11:49)
07. TaskManager. Part 1 (09:38)
08. TaskManager. Part 2 (10:12)
09. TaskManager. Part 3 (10:07)
10. TaskListViewController (09:04)
11. TaskListViewController. Part 2 (08:34)
12. DataProvider. Part 1 (08:12)
13. DataProvider. Part 2 (07:45)

14. Theory of Test Doubles (06:54)
15. MockTableView (10:23)
16. MockTableViewCell (05:32)
17. cellForRow method modification (07:08)
18. Refactor test sections (07:17)
19. Done & Undone button name tests (05:58)
20. Check & Uncheck functionality (08:22)
21. Check titleLabel in Cell (07:43)
22. Cell Configure Tests (06:16)
23. Done task style (08:27)
24. DetailViewController Elements (07:54)
25. Testing Labels on DetailViewController (11:42)
26. Date Label and MapView Tests (06:21)
27. Controller for New Tasks (05:35)
28. Getting Coordinates from Geocoder. Part 1 (13:55)
29. Getting Coordinates from Geocoder. Part 2 (08:26)
30. Test Geocoder Returned Coordinate (06:57)
31. Test Host URLComponent (11:18)
32. QueryItems Test (12:58)
33. Token Availability Test (16:39)
34. JSON Tests (05:57)
35. AddNewTask Button Tests (12:05)
36. Sharing TaskManager Test (05:36)
37. NewTaskViewController Dismiss Test (07:22)
38. TableView Reload Method Test (09:06)
39. NotificationCenter Test (06:22)
40. Push ViewController Test (11:22)
41. Model Serialization Tests (12:29)
42. Serialization into Dictionary Tests (08:05)
43. Save Data Test (13:42)
44. Checking Functionality (05:53)
45. Fixing Found Bugs (06:46)
46. UI Tests (16:04)
47. Coverage (02:34)
48. TDD. Unit Testing. Outro (00:48)

Общая продолжительность курса 7:09:14

22. VK News Feed App

Описание курса

Задумывались ли вы, как приложения создаются в реальном мире? Хотели ли вы попробовать себя в написании полноценного современного приложения? Давно планировали начать разрабатывать интерфейс приложения через код?

Представляем вам наш новый увлекательный курс, который будет посвящен созданию приложения по типу новостной ленты приложения ВКонтакте используя различные JSON APIs непосредственно из сервера ВКонтакте.

Мы реализуем огромное количество классного функционала в совершенно разных направлениях: от создания красивого пользовательского интерфейса двумя способами: через код и через XIB файлы до работы с сетью, запроса различных данных с сервера с помощью VK SDK API и парсинга JSON данных.

Чтобы по мере разрастания проекта код все также оставался понятным и читаемым в данном курсе я подниму вопрос архитектуры, и весь проект будет следовать архитектуре под названием Clean Swift.

Еще одна замечательная серия тем, которые мы затронем это я научу вас как динамически высчитывать размеры ячеек в зависимости от размера контента, который они содержат. И многое, многое другое!

Чтобы сделать уроки этого курса очень простыми, мы начнем с нуля, создав новый проект. Вот краткий список тем, который мы реализуем в этом курсе:

Clean Swift Architecture

User Authorization / Registration

JSON Decodable

Asynchronous Data Fetching

Auto Layout Programmatically / using XIB

NSLayoutConstraint and Anchor Extension

UITableViewCell Dynamic Behavior

Loading and Caching Images
Nested UICollectionViews
UICollectionView Custom Layout
UINavigationController
CAGradientLayer & Custom Shadows
Что вас ожидает:
30 видеуроков с подробным объяснением функционала
Исходный код для каждого урока
Swift 4.2 & 5 обновленный синтаксис

Содержание курса

01. Intro (08:04)
02. Installing VKSDK (15:39)
03. VK SDK and User Authorization (41:17)
04. VK SDK API Networking Logic (17:12)
05. Fetch Newsfeed JSON (11:10)
06. Implementing Parsing Logic (19:33)
07. The Importance of Architecture (09:24)
08. Setup Clean Swift Xcode Templates (03:37)
09. Clean Swift Architecture Overview (10:35)
10. Into Clean Swift VIP Cycle (14:55)
11. Newsfeed Cells Layout by XIB (23:11)
12. Parsing Data Into Cells (24:33)
13. Fetch Groups and Users JSON Data (19:35)
14. Loading and Caching Images (10:25)
15. Implementing Post Images JSON Fetching (28:00)
16. Cells Transformation With Card View (05:55)
17. Auto Sizing Cells Part 1 (24:07)
18. Auto Sizing Cells Part 2 (26:47)
19. Auto Sizing Cells Part 3 (17:57)
20. NSLayoutAnchor and Anchor Extension (24:47)
21. Auto Layout Programmatically Part 1 (21:17)
22. Auto Layout Programmatically Part 2 (38:29)
23. Cell Buttons Delegation (22:38)
24. Handling Cell Buttons (21:35)
25. Nested UICollectionViews (33:55)
26. UICollectionViewCustomLayout Part 1 (26:35)
27. UICollectionViewCustomLayout Part 2 (30:55)
28. Fetch and Show User Avatar Picture (30:48)
29. Manual Automatic Post Refresh and UI Improving (17:21)
30. Loading Previous Posts (37:13)
31. Previous Posts Footer Loader (18:14)
32. Gradient View and Cell Shadows (15:42)

Общая продолжительность курса 11:11:25

23. Работаем с CloudKit, iCloud

Описание курса

Данный курс является продолжением курса по UITableView, на котором мы разрабатывали приложение Places. Это приложение позволяет регистрировать и сохранять различные ваши места и заведения в локальной базе данных, в качестве которой мы использовали Realm. На этом курсе мы продолжим работу над этим приложением и реализуем возможность хранения данных в iCloud, что в свою очередь позволит синхронизировать все ваши данные между всеми устройствами в пределах одной учетной записи. Т.е. наша задача будет не просто сохранить запись в облачное хранилище, а именно обеспечить полную синхронизацию облачных данных с локальными.

Для прохождения этого курса вам понадобится оплаченная лицензия разработчика!

Содержание курса

01. Intro (01:24)
02. Connection iCloud and CloudKit (13:03)
03. Pass Data to iCloud (22:00)
04. Fetch Data from iCloud (16:07)
05. Auto Generation of a Unique ID (12:24)
06. Record ID (08:29)
07. Update Records in iCloud (07:19)
08. Query Optimization. Part 1 (08:31)

09. Downloading Images (10:03)
10. Bug Fixes (09:03)
11. Query Optimization. Part 2 (05:59)
12. Delete Record from Cloud (12:27)

Общая продолжительность курса 2:06:44

24. Фреймворк SwiftUI

Описание курса

Ну что, ребята, вы готовы начать изучать SwiftUI?

Каждый iOS разработчик обязан идти в ногу со временем, постоянно быть в курсе всех новых нововведений, которые вводят нам Apple и стараться как можно быстрее все изучить, так вот совсем недавно на WWDC 2019 парни из Купертино презентовали миру новый фреймворк под названием SwiftUI.

И с момента релиза мы не теряли ни дня и изучали этот фреймворк, чтобы первыми рассказать вам о нем, показать как с ним взаимодействовать и как его использовать.

Поэтому я рад вам представить наш новый курс, в котором мы подробно обзвеем SwiftUI! В этом курсе мы напишем приложение с нуля до готового проекта, интерфейс которого полностью напишем через новый UI Framework.

Этот курс бесплатный и выложен как на сайте, так и на нашем YouTube канале, поэтому приглашаю всех пройти его как можно скорее!

Содержание курса

01. SwiftUI- SwiftBook App Course Preview (06:33)
02. SwiftUI - Lists (26:44)
03. SwiftUI - Complex Layouts (22:07)
04. SwiftUI - How to do Navigation Between Screens (19:47)
05. SwiftUI - UIKit with SwiftUI Integration (07:03)

Общая продолжительность курса 1:22:13

25. Принципы S.O.L.I.D

Описание курса

Вы можете безгранично разбираться в любой технологии, знать бесчисленное количество фреймворков, но этого не достаточно, чтобы хорошо писать качественный код, в котором вы сами сможете разобраться даже взглянув на него после полугодового перерыва.

В этом курсе речь пойдет не об архитектурах, а об основополагающих принципах, на основе которых строятся архитектуры - принципах SOLID.

Вот как расшифровывается акроним SOLID:

S: Single Responsibility Principle (Принцип единственной ответственности).

O: Open-Closed Principle (Принцип открытости-закрытости).

L: Liskov Substitution Principle (Принцип подстановки Барбары Лисков).

I: Interface Segregation Principle (Принцип разделения интерфейса).

D: Dependency Inversion Principle (Принцип инверсии зависимостей).

Эти принципы были представлены Робертом Сесил Мартином, также известным в интернете как Дядюшка Боб. Дядюшка Боб - известная личность в мире программирования, и он же является основоположником чистой архитектуры, на основе которой и существуют известные архитектурные подходы при создании ПО при программировании ООП-приложений. Поэтому, изучив эти идеи, вы без труда сможете легко проникать в суть и смысл любой архитектуры, а может быть даже решите создать свой собственный архитектурный подход!

Какую проблему решают SOLID принципы?

Следуя данным принципам вы получите простой и понятный код, в который легко будет вносить изменения, поддерживать, а также работать над этим кодом в команде.

Применяя SOLID принципы на практике вы выведете структурность вашего мышления на новый уровень, что позволит вам самим браться за более сложные проекты без боязни запутаться в них. Знание и умение применять принципы SOLID на практике даст вам большой плюс при трудоустройстве, так как это подчеркнет, что вы не просто пишете код, а пишете его грамотно, и что это для вас действительно важно.

Чему вы научитесь в этом курсе?

- Изучим теоретическую составляющую каждого принципа SOLID
- Разберем применение каждого принципа на практических примерах
- Посмотрим как применяются SOLID принципы в рамках реального приложения
- Научимся думать перед тем, как бросаться писать код.

В качестве небольшого бонуса мы так же с вами разберем еще пару акронимов, которые так же пригодятся нам при написании качественного кода!

Содержание курса

01. Intro (01:37)
02. SOLID Principles (06:50)
03. Single Responsibility Principle (06:10)
04. Example - Typical UIViewController (20:56)
05. Violation of the SRP (07:33)
06. Big Project SRP Example (14:39)
07. Open Closed Principle (09:01)
08. Example - Network Logic (22:41)
09. Example - Implementing Generics in OSP (21:45)
10. Liskov Substitution Principle (04:23)
11. Proper Inheritance Rules (09:48)
12. Violation of the LSP (05:13)
13. Interface Segregation Principle (04:24)
14. Example - Fat interface (Protocol) (06:56)
15. Example - Fat interface in Class (12:03)
16. Dependency Inversion Principle (04:59)
17. Implementing abstraction in DIP (07:50)
18. Don't get trapped by S.O.L.I.D. (03:49)
19. DRY, KISS, YAGNI Principles (05:25)
20. Outro (00:33)

Общая продолжительность курса 2:56:27

26. Apple Music с использованием SwiftUI и UIKit

Описание курса

А хочется ли вам научиться создавать полноценные работоспособные приложения, которыми вы пользуетесь каждый день? Если да, то мы рады представить вам наш новый курс, в котором мы будем с нуля создавать приложение аналогичное Apple Music.

Курс будет насыщен всем тем функционалом, который вы так привыкли видеть в настоящих приложениях по работе с музыкальными файлами, а именно поиск и воспроизведение треков из iTunes API, сохранение понравившихся треков в память устройства с помощью UserDefaults и многое другое.

Также одной из основных особенностей курса будет заключаться в том, что он будет написан с использованием двух фреймворков: UIKit и SwiftUI, а именно покажу как внедрять целый экран, написанный через SwiftUI в приложение, полностью написанное на UIKit.

Приложение получится достаточно объемным, поэтому в той части проекта, которая будет написана через UIKit мы применим архитектуру под названием Clean Swift, чтобы максимально эффективно распределить задачи между файлами.

Огромное внимание уделим работе с анимациями, и реализуем кастомный анимированный переход к экрану воспроизведения трека, как в настоящем приложении Apple Music.

Разбирая курс по кусочкам, реализация приложения получится не сложной и интересной.

Создавая проект с нуля, мы затронем огромное количество и других различных технологий, которые указаны ниже:

1. Используем Alamofire библиотеку для запроса данных из сети
2. Поиск треков через iTunes API
3. AVKit Audio Player для воспроизведения музыкальных файлов
4. SDWebImage/UIKit и UIImage/SwiftUI библиотеки для загрузки изображений
5. Создаем анимированные переходы через Auto Layout
6. Учимся работать с UIPanGestureRecognizer/UIKit, LongPressGesture/SwiftUI
7. Сохраняем понравившиеся треки в память устройства через UserDefaults
8. Используем UIKit и SwiftUI в рамках одного проекта
9. Перетаскиваемый Track Detail Player доступен между всеми экранами приложения
10. Декодируем входящий JSON ответ в асинхронном потоке
11. Применяем архитектуру Clean Swift

Что вас ожидает:

- 30 видеоуроков с подробным объяснением функционала
- Исходный код для каждого урока
- Swift 5. При любых изменениях в синтаксисе курс будет обновлен до актуальной версии

Содержание курса

01. Intro (06:08)
02. MainTabBarController (15:54)
03. Setup Search UITableViewController (13:17)
04. Alamofire and iTunes Search API (13:13)
05. Parsing logic using Decodable (18:42)
06. Smart Refactoring Patterns (14:58)

07. Setup Clean Swift Templates (09:18)
08. Clean Swift Architecture Explained (22:58)
09. Clean Swift VIP Cycle First (19:54)
10. Creating Track Cells by XIB (25:58)
11. Loading and Caching Images with SDWebImage (06:36)
12. Search Delay Fixing (21:16)
13. Track Detail Controller (09:33)
14. Track Detail Design Continued (19:30)
15. Play Tracks with AVPlayer (12:35)
16. Scale Animations and Time Observer (14:25)
17. Tracking Playback Time (10:22)
18. Time Slider and Volume Slider (09:46)
19. Go Forward and Go Back Buttons (20:14)
20. Track Player Transition Animation (22:34)
21. Play Track Maximizing Animation (14:20)
22. Mini Player View (18:32)
23. Drag and Drop UIPanGesture (19:52)
24. Dismissal UIPanGesture (08:43)
25. Library View SwiftUI Setup (23:58)
26. Save Tracks with UserDefaults (23:56)
27. Dynamic List with Tracks (20:53)
28. SwiftUI Gestures and Custom Buttons (23:37)
29. Improve Delegate Logic (15:41)
30. Outro (01:24)

Общая продолжительность курса 7:58:07

I уровень - Основы языка Swift

Урок 1

- Знакомство с группой
 - Краткое описание курса
 - Регламент уроков
 - Краткая история Swift
 - Почему Swift?
 - Почему надо изучать Swift?
- Синтаксис Swift, основные концепции
- Знакомство с Playground
 - Константы и переменные
 - Базовые типы или типы данных
 - Приведение типов

Урок 2

- Базовые операторы
- Арифметические операторы
 - Операторы присваивания
 - Операторы сравнения
 - Операторы диапазона
 - Логические операторы
- Строки и символы
- Работа с символами
 - Конкатенация строк и символов
 - Интерполяция строк

Урок 3

- Условные инструкции
- If
 - Тернарный оператор
 - Switch

Опционалы

Урок 4

- Типы коллекций
- Массивы
 - Словари
 - Множества
- Циклы
- for-in

- while
- repeat-while

Урок 5

Функции

- Определение функции
- Объявление функции
- Параметры функций и возврат значений
- Вложенные функции (Ознакомление)

Замыкания

- Определение замыкания
- Функции как замыкания
- Замыкающие выражения
- Переменные-замыкания
- Захват значений замыканиями

Урок 6

ООП в Swift

- Общее понятие ООП
- Классы и объекты
 - Описание класса
 - Свойства класса
 - Виды методов
 - Экземпляры класса
- Наследование
- Инкапсуляция
- Полиморфизм

Урок 7

Перечисления

- Описание перечисления
- Синтаксис перечислений
- Возможности перечислений

Структуры

- Описание структуры
- Сравнение классов и структур
- Методы в структурах

Урок 8

Дополнительные концепции Swift:

- КORTEЖИ
- Оператор guard

II уровень - Мои первые приложения на Swift

Урок 1. Среда разработки Xcode. Работа с Git

Библиотека объектов

Знакомство с интерфейс билдером

- Учимся проектировать интерфейс на примере приложения Hello World.
 - Стандартные шаблоны приложения
 - Storyboard
 - UIViewController
 - Элементы интерфейса (UIButton, UILabel и т.д.)
 - AutoLayout
- Расстановка элементов интерфейса при помощи констрейнтов
- Объединение элементов интерфейса в StackView
- Соединяем элементы интерфейса с кодом

Работа с Git

Урок 2. UIKit

- UISegmentedControl
- UILabel
- UITextField
- UIButton
- UISlider
- UISwitch
- UIScrollView
- UIDatePicker

Урок 3. Контроль и приведение типов

Переходы и контроллеры навигации

Контроль и приведение типов

- Контроль типов
- Приведение типов
- Работа с типом Any

Navigation Controller

- переходы между контроллерами представлений
- создание переходов
- возврат в исходное состояние
- добавление и настройка контроллера навигации
- навигационная панель
- элементы навигации
- увеличение заголовка
- передача информации между контроллерами
- создание переходов программным способом

Tab Bar Controller

- когда использовать контроллер панели вкладок
- добавление контроллера панели вкладок
- добавление вкладок
- элементы панели вкладок
- настройка элементов панели вкладок
- больше вкладок — появление элемента «More»
- добавление контроллеров представлений в контроллер панели вкладок

Урок 4-5. Пошаговый проект: Personality Quiz - персонализированный опросник

- создание интерфейса
- добавление вопросов и ответов
- модель данных и соединение интерфейса с кодом
- вычисление и отображение результатов

Урок 6. UITableView

- UITableViewController
- UITableView
- UITableViewDataSource
- UITableViewDelegate
- UITableViewCell

Урок 7. Протоколы

- CustomStringConvertible
- Equatable
- Comparable
- Codable

Создание собственного протокола

Делегирование

Урок 8. Анатомия приложения. Архитектура MVC

Жизненный цикл UIViewController

- viewDidLoad
- viewWillAppear
- viewWillLayoutSubviews
- viewDidLayoutSubviews
- viewDidAppear
- viewWillTransitionToSize
- viewWillDisappear
- viewDidDisappear

Жизненный цикл приложения

- didFinishLaunchingWithOptions
- applicationWillResignActive
- applicationDidEnterBackground
- applicationWillEnterForeground
- applicationDidBecomeActive
- applicationWillTerminate

Схема MVC

Определение Model, View, Controller

Приложение MVC Example

Урок 9. CocoaPods и работа с анимациями

- Понятие Cocoa
- CoreAnimation
- Пример кода для базовой анимации
- Понятие CocoaPods и фреймворков
- Работа с CocoaPods в терминале

- Интеграция CocoaPods в проект
- Фреймворк Spring для работы с анимацией

Урок 10. Сетевые запросы

URLSession

- Знакомство с API URLSession
- HTTP запросы
- URLSessionDataTask
- Загрузка изображений из сети

JSON

- Анатомия JSON формата
- Модель данных
- Парсинг JSON
- JSONDecoder
- Обработка ошибок
- Конвертация формата snake_case в camelCase

Урок 11. URLSession vs Alamofire

URLSession

- From snake_case to camelCase
- POST Request

Alamofire

- Знакомство с фреймворком Alamofire
- Парсинг JSON с использованием Alamofire
- Status Code
- Валидация данных
- responseJSON
- responseData
- responseString
- response
- POST Requests

Урок 12. User Defaults. Plis СПАМ Image Caching

UserDefaults

- Знакомство с классом
- Примеры сохранения и восстановления данных
- Работа с моделями

plist File

Image Caching

Урок 13. CoreData

- Подготовка приложения для работы с CoreData
- Знакомство с фреймворком
- Сохранение данных
- Восстановление данных

Урок 14. Пошаговый проект: ToDo List с использованием Realm

- Знакомство с фреймворком Realm
- Создание моделей для хранения данных в Realm
- Добавление объектов в базу
- Запрос к базе данных
- Отображение данных в интерфейсе приложения
- Обновление данных

Урок 15. Realm. Часть 2

- Добавление данных в базу через интерфейс приложения
- Редактирование данных
- Удаление данных
- Применение фильтров
- Сортировка
- Кастомизация интерфейса приложения

Урок 16. Работа с памятью

- Memory Basics
- Finding and Fixing Memory Leaks
- Types of References
- Retain Cycles in Closures