

"Powiedz mi, a zapomnę.
Pokaż mi, a zapamiętam.
Pozwól mi zrobić, a zrozumiem." - Konfucjusz

Twisted Academy - pomożemy Ci zrozumieć, jak tworzyć aplikacje na iOS

Oferujemy:

- Intensywny kurs, w trakcie którego własnoręcznie stworzysz swoją pierwszą aplikację na iPhone'a i w pełni zrozumiesz wszystkie aspekty tego procesu.
- Nauczysz się posługiwać wszystkimi narzędziami niezbędnymi do tworzenia i publikacji aplikacji mobilnych.
- Poznasz rozwiązania wielu typowych problemów, które występują podczas programowania na iOS
- Dowiesz się, skąd czerpać wiedzę, by dalej rozwijać swoje umiejętności
- Uzyskasz dostęp do forum dla absolwentów kursu (możliwość uzyskania indywidualnego wsparcia od prowadzących i pozostałych uczestników)

Wymagania:

- Chęć tworzenia aplikacji na iOS
- Umiejętność programowania w dowolnym języku obiektowym na poziomie zaawansowanym (czyt.: zrozumienie takich pojęć jak: class, instance, method, member variable, class variable, static, abstract, synchronous, asynchronous, callback, modal)
- Umiejętność poruszania się w OS X
- Macbook lub Mac Mini z zainstalowanym XCode'em

Formuła

- Szkolenie jest przeznaczone dla doświadczonych programistów, którzy chcą postawić pierwsze kroki w programowaniu na platformę iOS
- Uczymy nie tylko jak tworzyć aplikacje, lecz - przede wszystkim - jak uczyć się je tworzyć
- Kładziemy nacisk na praktykę, a nie teorię - przekazujemy bardzo dużo bardzo praktycznej wiedzy (zasada Pareto: 20% wiedzy - 80% możliwości)
- Zakładamy aktywny udział - każdy uczestnik programuje przy komputerze, a nie tylko słucha
- Małe grupy, indywidualne podejście

Szczegółowy program:

Dzień 1

I. Wstęp (1h)

- Omówienie aplikacji do stworzenia (TabBar, NavBar, Controller, Modal)
- Utworzenie projektu
- Poznanie wszystkich elementów projektu oraz interfejsu XCode'a (Navigators, Utilities, Interface Builder)
- Poznanie symulatora iOS i uruchomienie w nim aplikacji
- Commitowanie projektu w Gicie

II. Budowa interfejsu (1h)

- Dodanie pierwszego ekranu (label, button, outlet)
- Dodanie drugiego ekranu (IBAction event, segue)
- Dodanie nawigacji Master-Detail (UINavigationController, segue, push, modal)
- Dodanie TableView (UITableViewController, UITableView, UITableViewCell)
- Dodanie UIImageView

III. Pierwsza krew, czyli zaczynamy kodować (6-8h)

- [Podstawy składni Obj-C](#)
- Struktury danych (stringi, tablice, słowniki itp.)
- Wyświetlanie dynamicznych danych na ekranie
- Obsługa zdarzeń
- Przekazywanie danych z Master do Detail
- Zapisywanie danych (TextField, protocol, delegate)
- Asynchroniczne pobieranie obrazków ([categories](#))

Dzień 2

IV. Pobieranie danych z API (3-4h)

- Pobieranie i mapowanie danych z API
- Wysyłanie danych do API

V. Wyższa szkoła jazdy (czas: 3-4h)

- Edycja danych
- Zapisywanie danych - UserDefaults, CoreData
- Nakładanie grafiki

VI. Publikacja aplikacji na potrzeby testów (2h)

- Provisioning Portal
- Ustawienia projektu (nazwy, splashe, ikony)
- Wrzucenie aplikacji na telefon (code signing, provisioning portal)
- Udostępnianie aplikacji testerom (TestFlight)

Potencjalne dalsze kroki (2. stopień kursu)

- Geolokalizacja (Core Location)
- Dodanie zdjęcia z kamery/galerii
- Dostosowanie aplikacji do iPada
- Mapy (MapKit)
- Publikacja via FB/Twitter
- Publikacja aplikacji w AppStore
- Ekran logowania
- (ShareKit)
- ...

Poruszone tematy:

- Poruszanie się w XCode
 - Tworzenie projektu
 - Organizacja plików
 - Navigators, Utilities
 - Interface Builder
 - Storyboards
 - Kontrolki
 - Kontrolery
 - Atrybuty
 - Outlety
 - Nakładanie grafiki
 - Ustawienia projektu
 - XCAssets (Splashe, ikonki)
 - Targety, Schemes, info.plist
 - Certyfikaty, podpisywanie
 - Frameworki Apple (Cocoa Touch, Core Data, Core Location, MapKit)
 - Frameworki zewnętrzne (RestKit, APIRater, TestFlight, Flurry, DataKit, FB)
 - Debugging
 - NSLog, NSAssert
 - Breakpoints (manual, exception, symbolic)
 - lldb (po, p, bt, watch set variable variable_name)
 - Opcja NSZombieEnabled
 - iOS Simulator
 - Wybór urządzenia i wersji iOS
 - Skalowanie i animacje
 - Reset
 - Organizer
 - Archive
 - Profiles
 - Integracja z Git

- Podstawy Objective-C:
 - I
 - Składnia
 - Interfaces, implementations
 - Messages, Selectors
 - Protocols
 - Delegates
 - Properties, ivars
 - NSString, NSArray NSDictionary, NSNumber/NSInteger (+Mutable)
 - alloc, init, initWith.../...With...
 - Literals
 - Primitives types vs. objects (%@, %d, %f itp.)
 - Casting
 - Zarządzanie pamięcią (ARC)
 - Preprocessor directives (#define, #ifdef, #pragma mark)
 - II
 - Blocks
 - Categories
 - Notifications
 - NSUserDefaults
 - NSURLConnection
 - i18n
 - Push Notifications
 - KVC, KVO
- Zasady projektowania aplikacji
 - iOS Human Interface Guidelines
 - App Review Guidelines
- Publikacja aplikacji w App Store
 - iOS Provisioning Portal
 - Certyfikaty (Development, Distribution, Ad-Hoc)
 - Provisioning Profiles
 - Przygotowanie aplikacji tak, by nie została odrzucona
 - Wrzucenie aplikacji do iTunes Connect
- Narzędzia zewnętrzne
 - Git
 - TestFlight
 - Crashlytics
 - Parse.com
 - FireBase
 - Flurry
- Gdzie zdobywać dalsze informacje (developer.apple.com, stackoverflow.com, iphonedevsdk.com itp., zamknięte forum, drugi poziom kursu)