

Course code: **IOS**

Course title: **iOS platform programming**

Days: **5**

## Description:

### Course intended for:

The training is intended for programmers, who want to get familiar with Objective-C language and iOS platform from the start, including the more advanced issues (such as working with a large database).

### Course objective:

The training objective is to get the participants familiar with the iOS platform and with the requirements defined by Apple company for development of applications for iPhone, iPod Touch and iPad devices. The training curriculum is focused on the solid theoretical base (understanding of documentation and techniques, including a historic outline, e.g. methods of manual memory management) with practical applications (based on the example of popular and featured applications in the AppStore).

During a 5-day course, the participants will get to know the Objective-C language and the Apple (Xcode IDE) environment, popular patterns used in the iOS platform, and they will get a broad review of several available frameworks thanks to development of several applications, using different tools and techniques.

### Course strengths:

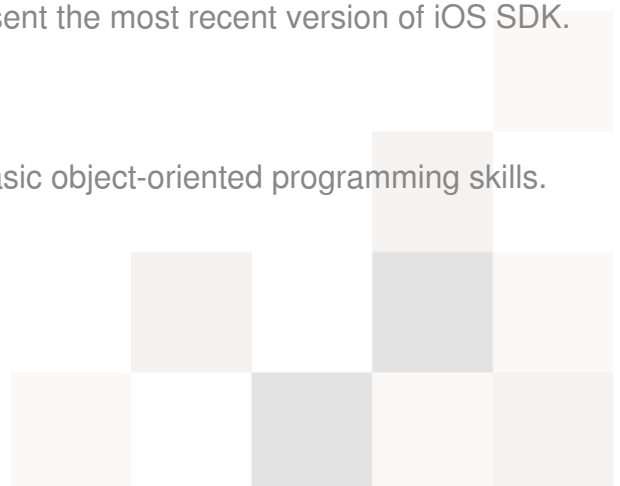
The curriculum encompasses the issues of development of mobile applications for iOS platform in the Objective-C language. Not only the basic topics will be discussed, but also those, which are used in many applications, and their learning on one's own may turn out to be rather problematic.

The curriculum is constantly updated to always present the most recent version of iOS SDK.

### Requirements:

The training participants are required to have the basic object-oriented programming skills.

### Course parameters:



5\*8 hours (5\*7 net hours) of lectures and workshops (workshops constitute 80% of the training curriculum).

During the workshops, applications using the most frequently applied frameworks in the popular iOS applications are developed.

## Course curriculum:

### 1. Basics and introduction to iOS SDK

I. The historic outline of NextStep and development of Apple iOS in the recent years

II. History of iPhone and iPad devices

III. Software and hardware requirements

IV. Discussion of iOS developer programs:

Free account

Student account

Individual account

Corporate account

Enterprise program

V. Programming tools

Xcode and Interface Builder,

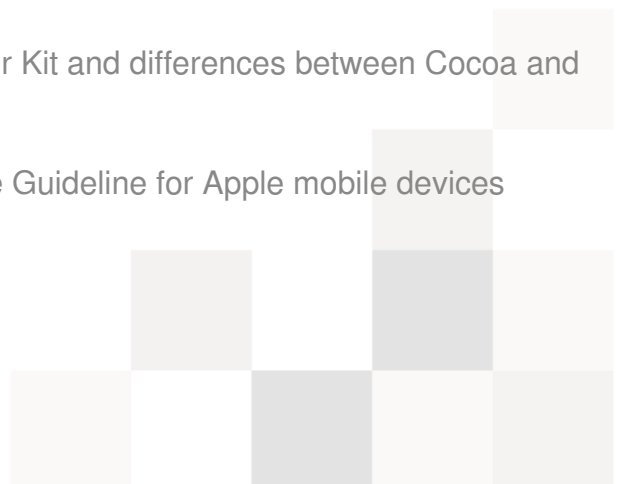
iOS Simulator,

Instruments

VI. An outline of iOS Software Developer Kit and differences between Cocoa and Cocoa Touch

VII. Getting to know the Human Interface Guideline for Apple mobile devices

Best practices



Potential reasons for rejection of the application upon validation

Case Studies – examples of Polish companies

## VIII. Programming in Objective-C language

Objective-C in comparison with C language

Language syntax and presentation of new types of data

Sending of messages

Classes and instances

The role of NSObject in memory management

Memory management methods (manual and ARC)

NSArray and NSDictionary classes

Development of strings, comparison of write lines etc.

Creation of numerical objects

Creation of getters and setters using @property

Creation of own getters and setters

Using the notification center

Using the delegation pattern

Using the singleton pattern

Using the facade pattern

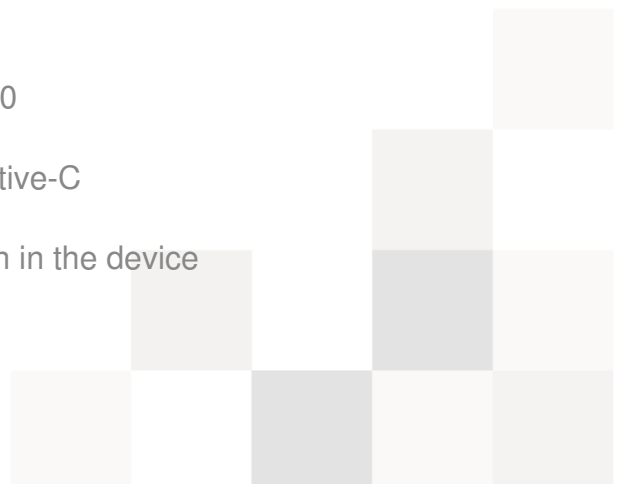
Defining and management of protocols

Defining of categories

What's new in Objective-C 2.0

What's new in Modern Objective-C

## IX. Application launching and distribution in the device



Generating of the required certificates and data profiles

Distribution using TestFlightApp

Distribution to AppStore (Distribution)

Enterprise Distribution

## 2. Views and navigation between views in iOS

### I. application building and structure

Discussing of NSBundle

.h, .m, .xib, .nib, .plist files

Development of objects in the Interface Builder

The Model-View-Controller paradigm in iOS SDK

Presentation of IBAction and IBOutlet

UIWindow class

UIView class

UIButton class

UIResponder class

### II. Photo browser building, overriding the Interface Builder

Using of the parallax effect

Getting familiar with the UIScrollView control

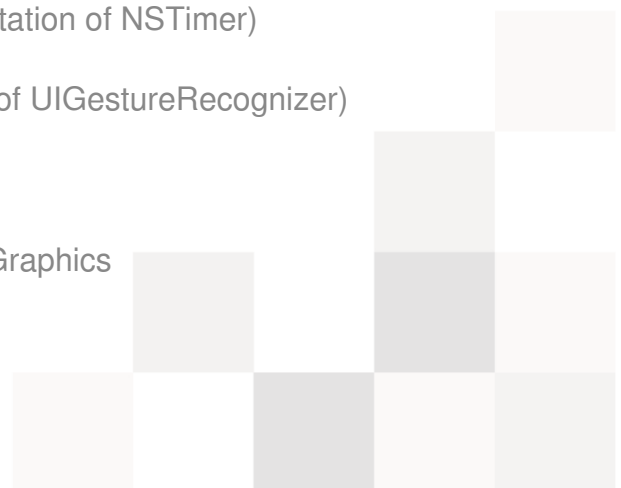
Use of the UIScrollViewDelegate protocol

Automatic slide view (presentation of NSTimer)

### III. User gesture capturing (subclasses of UIGestureRecognizer)

Basics of CoreGraphics

Essential structures of CoreGraphics



## Effective use of macros

### IV. Interface structure:

Application building using StoryBoard

Use of AutoresizingMask to build simple interfaces

Building of interfaces using AutoLayout

Building of interfaces using the code only

Building of interfaces for iPhone 4/4S and iPhone 5

Building of interfaces for iPad and iPad mini

Development of universal applications

Development of multilanguage applications

Translation of texts and graphics

Translation using Base Internationalization

### V. Navigating between many views

Presentation of UINavigationController

Presentation of UITabBarController

### VI. Structure of the Web browser

Using the UIWebView control

Use of local Web pages

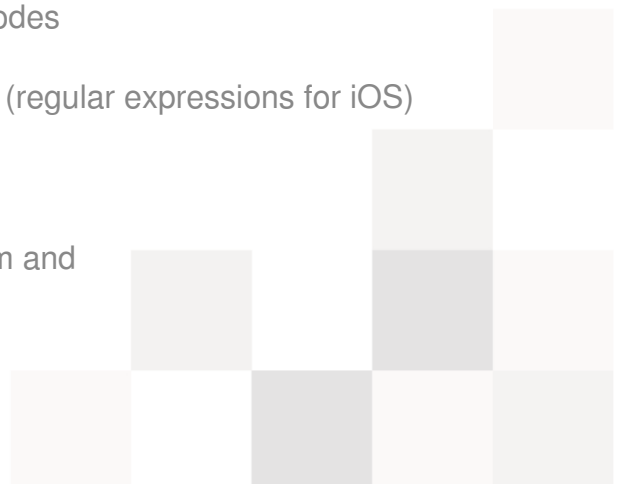
Displaying tutorials using HTML

Performance of JavaScript codes

Validation of Web addresses (regular expressions for iOS)

### VII. Project integration with CocoaPods

## 3. Development of lists and the local file system and



## I. File management in the iOS system

Permanent data storage

Presentation of NSFileManager singleton

Saving of settings and development of global settings

Serialization of objects and shortcomings of individual solutions

Data storage security

Handling of data in JSON, XML and plist formats

## II. Adding of UISearchBar and UISearchBarController

## III. Adjusting of UITableView

Management of UITableViewDelegate and UITableViewDataSource protocols

Differentiation of default styles

Development of indexes (ABC... like in an address book)

Table sorting

Custom style creation

Advanced cell development methods

Differences between iOS versions

## IV. Use of UIMapView and CoreLocation

User location identification on the basis of GPS

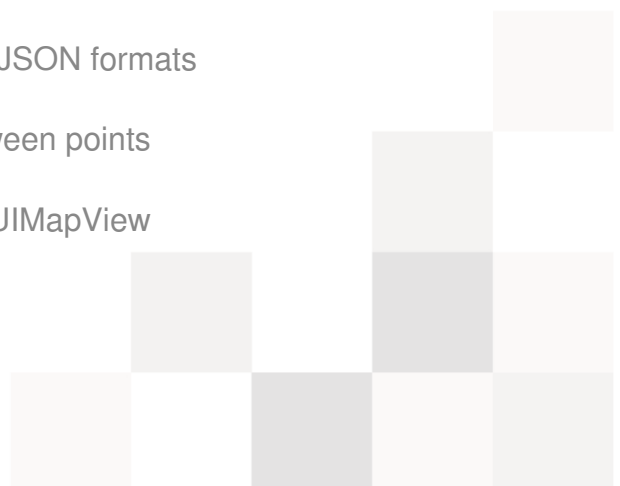
Location simulation in iOS Simulator

Presentation of gpx and GeoJSON formats

Calculating of distances between points

Entering components in the UIMapView

## 4. Concurrent programming



## I. Creation and handling of threads in iOS system

## II. Blocks- Introduction to anonymous systems

Ways of asynchronous code performance

Development of blocks and memory allocation

Memory management and ways to avoid the Strong Retain Cycle

## III. Introduction to Grand Central Dispatch (GCD)

Synchronous queues

Asynchronous queues

Defining of operations for performance of tasks

Analogies to manual thread management

## IV. Discussing of NSOperationQueue and NSOperation classes

The proper use of NSBlockOperation class

NSOperation subclasses with the discontinuation option

Concurrent operations performance

## V. Downloading of data

Discussing of NSURL,NSURLRequest and NSURLConnection classes

Discussing of the NSURLConnectionDelegate protocol

Downloading of data using NSURLConnection and completionBlock

Sending of requests to POST/PUT/DELETE

The basics of AFNetworking

Debugging of Web connections

## 5. Information storage in the database

### I. Introduction to CoreData



- II. CRUD methods
- III. Development of subclasses for objects
- IV. Development of links between objects
- V. Development of objects using categories
- VI. Import and export of large quantities of data
- VII. Generating of exemplary data
- VIII. Data search
  - Introduction of NSPredicate class
  - Combining and sorting of queries
- IX. Presentation of NSFetchedResultsController protocol
- X. Optimization for very large data sets
- XI. Introduction of changes in database structure (Schema edition)
- XII. Automatic migrations
- XIII. Development of many contexts using the Parent-Child paradigm
- XIV. Discussing of popular CoreDataStack solutions

