

Course code: **IOS/START**

Course title: **iOS programming – a quick start**

Days: 3

## Description:

### Course intended for:

The training is intended for programmers, who want to get to know the Objective-C language and the iOS platform from the start, including more advanced issues (e.g. 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. Three complete applications will be written during the training.

During a three-day course, the participants will get familiar with the Objective-C language and the Apple developer environment (Xcode IDE), the popular patterns used in the iOS platform and several accessible frameworks.

### Course strengths:

The training curriculum includes mobile application development for iOS platform in the Objective-C language. Not only the basic topics will be presented, but also those, which are used in many applications and can be difficult to learn on one's own.

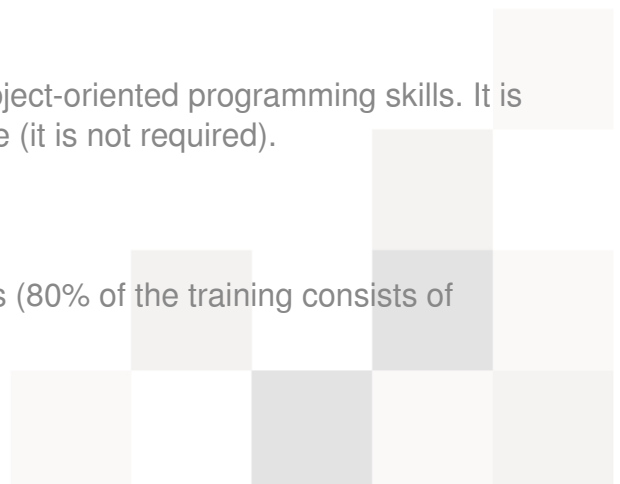
The curriculum is being constantly updated to present always the new version of iOS SDK.

### Requirements:

The training participants are required to have the object-oriented programming skills. It is recommended that they are familiar with C language (it is not required).

### Course parameters:

3\*8 hours (3\*7 net hours) of lectures and workshops (80% of the training consists of



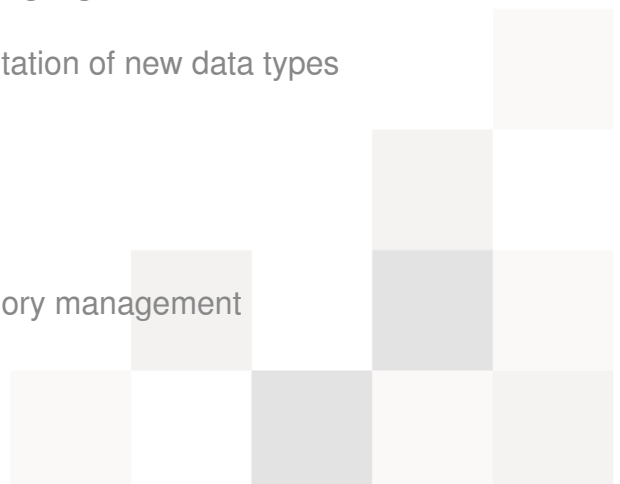
workshops).

During the training, the participant will develop applications using the most frequently utilized frameworks in popular iOS applications.

## 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. Hardware and software requirements
- IV. Programmer tools
  - i. Xcode and Interface Builder,
  - ii. iOS Simulator,
  - iii. Instruments
- V. The outline of iOS Software Developer Kit and differences between Cocoa and Cocoa Touch
- VI. Getting to know the Human Interface Guideline for Apple mobile devices
  - i. Best practices
  - ii. Case Studies illustrated by the example of Polish companies
- VII. Programming in Objective-C language
  - i. Objective-C in relation to C language
  - ii. Language syntax and presentation of new data types
  - iii. Message transmission
  - iv. Classes and instances
  - v. The role of NSObject in memory management



- vi. Memory management methods (manual and ARC)
- vii. NSArray and NSDictionary classes
- viii. Development of strings, comparison of write lines etc.
- ix. Development of numeric objects
- x. Development of getters and setters using property
- xi. Development of own getters and setters
- xii. Using the delegation pattern
- xiii. Using the singleton pattern
- xiv. Defining and management of protocols
- xv. Defining of categories
- xvi. New aspects of Modern Objective-C

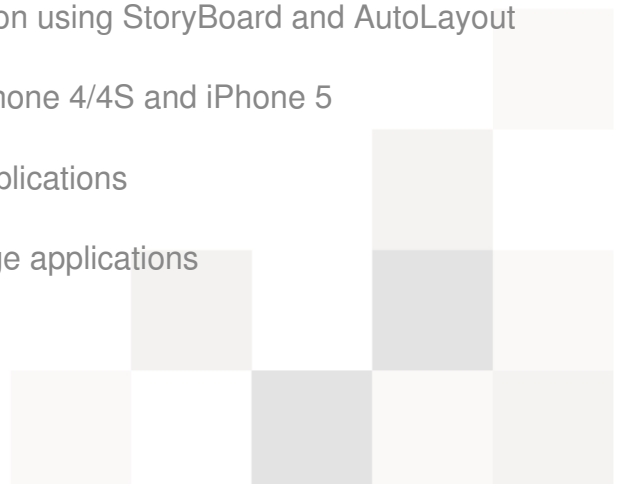
## 2. Views and navigation between views in iOS

### I. Application development and structure

- i. Discussing of NSBundle
- ii. .h, .m, .xib, .nib, .plist files
- iii. Development of objects in Interface Builder
- iv. The Model-View-Controller paradigm in iOS SDK
- v. Presentation of IBAction and IBOutlet

### II. Interface structure:

- i. Development of the application using StoryBoard and AutoLayout
- ii. Interface development for iPhone 4/4S and iPhone 5
- iii. Development of universal applications
- iv. Development of multilanguage applications



- v. Translation of texts and graphics
- vi. Translation using Base Internationalization

### III. Navigation between many views

- i. Presentation of UINavigationController
- ii. Presentation of UITabBarController

### IV. File management in the iOS system

- i. Presentation of the NSFileManager singleton
- ii. Security of data stored
- iii. Data management in JSON formats

### V. Adapting of UITableView

- i. UITableViewDelegate and UITableViewDataSource protocols
- ii. Development of indexes (ABC... like in an address book)
- iii. Adding of UISearchBar and UISearchBarController

### VI. Project integration with CocoaPods

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

### I. Concurrent programming

- i. Blocks - Introduction to anonymous functions
- ii. Discussing of the NSOperationQueue class
- iii. Asynchronous data collection

Discussing of NSURL,NSURLRequest and NSURLConnection

Downloading of data using NSURLConnection and completionBlock

Sending of requests POST/PUT/DELETE

Basics of AFNetworking

## II. Basics of databases for iOS

- i. Introduction to CoreData
- ii. CRUD methods
- iii. Development of subclasses for objects
- iv. Development of associations between objects
- v. Data search
- vi. Presentation of NSFetchedResultsController protocol
- vii. Introduction of changes in the database structure (Schema edition)
- viii. Automatic migrations
- ix. Development of many context using the Parent-Child paradigm

