

## CONTEÚDO PROGRAMÁTICO - TREINAMENTO BÁSICO EM IOS

Esse treinamento é direcionado aos profissionais que dominam lógica de programação ou desenvolvedores que ainda não possuem experiência com desenvolvimento mobile.

### 1. Entendendo o ecossistema de desenvolvimento

**Objetivo:** Entender os requisitos necessários para um projeto iOS.

- Certificados e Provisioning Profiles
- Processo de compilação e publicação de apps
- Compilação para teste no simulador e device, empacotamento e distribuição adhoc, Distribuição na App Store, distribuição in-House

### 2. Meu primeiro aplicativo: Hello World

**Objetivo:** Iniciar na produção de software para a plataforma. Executar uma primeira versão de um aplicativo no simulador.

- Conhecendo a estrutura de um projeto para iOS
- Ambiente XCode

### 3. Objective-C

**Objetivo:** Conhecer a linguagem, sua sintaxe e convenções

- Tipos primitivos, ponteiros, id, seletores, mensagens entre classes e chamadas de funções
- Orientação a objetos: Interfaces, Implementações, Categorias e Protocolos
- Properties, Blocks
- Gerenciamento de memória
- Tratamento de exceções

Conhecer as classes básicas mais utilizadas do framework e ser capaz de trabalhar com estruturas de dados de coleções

- Classes básicas: NSNumber, NSDate, NSString, NSValue, NSData, NSArray e seus mutáveis
- Coleções: NSArray, NSSet, NSDictionary, e seus mutáveis e Enumerations.

### 4. Interface com usuário Parte 1

**Objetivo:** Entender a arquitetura do sistema de Views e camadas do UIKit.

Entender o sistema de notificação de eventos e posicionamento dos elementos de tela

Ser capaz de tratar eventos dos usuários

Empregar os componentes mais importantes disponibilizados pelo UIKit

- UIView+UIViewController (padrão MVC)
- Interação através de eventos de toque (UITouch, UIEvent)
- Interface Builder ◦ Nib files
- UIKit e principais componentes nativos: UITextField, UITextView, UISlider, UISwitch, UIButton, UILabel, UIImage, UIImageView

## 5. Interface com usuário Parte 2

**Objetivo:** Ser capaz de criar views customizadas e complexas.

Ser capaz de executar animações sobre elementos de tela

Dar suporte a tratamento de gestos avançados

- Containers
  - UITableViewController
  - UINavigationController
  - UITabBar
  - UIToolBar
  - UIScrollView
  - UIWebView
- Reconhecimento de gestos com UIGestureRecognizer
- Animações
- Transformações
- Noções de Core Animation
- Noções de Core Graphics
- Noções de OpenGL

## 6. Gerenciando e persistindo dados

**Objetivo:** Entender a diferença entre as várias opções de formatos de armazenamento de dados

Criar mecanismos para que os dados da aplicação sejam persistidos em vários formatos para posterior recuperação,

- NSUserDefaults
- Sistema de arquivos
- Banco de dados relacionais embarcados com SQLite
- CoreData
- PropertyLists
- Xml (NSXMLParser e outras libs)

## 7. Redes e conectividade

**Objetivo:** Ser capaz de se comunicar com serviços web, enviando e recebendo dados através de requisições http



- NSURL
- NSURLConnection
- NSURLRequest

## 8. Pcessamento assíncrono

**Objetivo:** Entender os diferentes recursos disponíveis para implementação de processamentos Assíncronos

Criar aplicações com suporte à execução em background

- Threads,
- Loops de execução
- Operações assíncronas, Timers e Fila de Operações
- Backgrounds tasks ○ Centro de notificações

## 9. Acelerômetro e Giroscópio

**Objetivo:** Conhecer o framework CoreMotion.

Criar aplicativos capazes de reagir ao movimento físico dos dispositivos.

## 10. Push notification

**Objetivo:** Entender o funcionamento do sistema APNS(Apple Push Notification Service) e implementação de um servidor de notificações para testes.

## 11. CoreLocation e MapKit (posicionamento por triangulação e GPS)

**Objetivo:** Ser capaz de coletar dados sobre a localização atual (exata ou aproximada) do dispositivo

Ser capaz de utilizar o MapKit para exibir mapas e informações geográficas contextuais.

## 12. Sistema multimídia

**Objetivo:** Entender a arquitetura do sistema multimídia do iOS

Executar conteúdo multimídia dentro das aplicações

- System AudioServices
- MediaPlayer framework
- Noções de HTTP LiveStream
- Noções de OpenAL