

# Desenvolvimento de Aplicações para iPhone

# Aula 01



## Introdução ao iOS

**Prof. Othon Marcelo Nunes Batista**  
Mestre em Informática

# Roteiro

- Objetivos
- Pré-Requisitos
- Atividades
- Arquitetura do iOS
- Componentes da Plataforma
- Primeira Aplicação

# Objetivos

- O principal objetivo deste curso é a implementação de aplicativos para o iOS.
- Para que este objetivo seja cumprido, precisamos:
  - saber programar orientado a objetos;
  - conhecer MVC;
  - ter um computador com Mac OS X;
  - aprender a usar o Xcode;
  - aprender a programar em Swift.

# Pré-Requisitos

- Experiência em programação, e preferencialmente orientada a objetos é obrigatória.
- Estes termos devem ser conhecidos:
  - classe;
  - instância;
  - mensagem;
  - método;
  - variável de instância;
  - superclasse/subclasse.

# Arquitetura do iOS



Núcleo do sistema operacional Unix

Mach 3.0

Unix BSD

Sockets

Segurança

acesso a chaves

certificados

Gerência de energia elétrica

Sistema de arquivos

Bonjour

# Arquitetura do iOS



Coleções

Núcleo da localização

Contatos

Rede

Threads

Acesso aos arquivos

Preferências

SQLite

Utilitários de URL

# Arquitetura do iOS



Áudio no núcleo

JPEG, PNG, TIFF

OpenAL

Mixagem de áudio

gravação de áudio

Quartz (gráficos em 2D)

PDF

Animação no núcleo

Playback de vídeo

OpenGL ES (gráficos em 3D)

# Arquitetura do iOS



Multi-toque

Alertas

Movimentação (giroscópio e acelerômetro)

Web view

Hierarquia de views

Controles

Map kit

Localização (GPS e Glonass)

Image Picker

Câmera



# Componentes da Plataforma

- Ferramenta



**Xcode**

- Linguagens



**Swift**

- Frameworks



**Foundation**



**UI Kit**

**Core Data**  
**Map Kit**

- Estratégia de Projeto



**Model, View, Controller**

# Primeira Aplicação

- Nós implementaremos uma calculadora para nos acostumarmos com Swift e Xcode.
- Esta e a próxima aula cobrirão esta aplicação do zero!
- Tópicos a serem cobertos na aplicação nesta aula:
  - criação de um projeto no Xcode;
  - construção de uma *interface* com o usuário com *autolayout*;
  - uso do simulador;
  - print com conversão utilizando `\()`;
  - definição de classe em Swift incluindo como especificar variáveis de instância e métodos;
  - conexão de propriedades (variáveis de instância) do código em Swift para a *interface* com o usuário (*outlets*);
  - conexão de elementos da *interface* com o usuário para chamar métodos no código em Swift (*actions*);
  - acesso da documentação do iOS a partir do código;
  - Optionals (operandos `?` e `!`).