

INTRODUÇÃO A PLATAFORMA iOS



BREVE HISTÓRICO

PRIMÓRDIOS

Lançado como sistema operacional do 1º iPhone em 2007.

Originalmente chamado de iPhoneOS.

Era exclusivo para o dispositivo e totalmente fechado.

Continha apenas alguns Apps nativos e trazia uma experiência de uso diferenciada para a interação Multi-Touch capacitiva.

Curiosidade: ironicamente a Apple defendia nas primeiras versões do iOS que o futuro seriam os Web Apps, graças a experiência que eles conseguiram proporcionar com o Safari no primeiro iPhone.

VERSÕES DE iOS

iPhoneOS 1 (2007)

Lançamento do iPhone;
Recursos básicos de
Smartphone: ligação, SMS,
Emails e Browser.

iPhoneOS 2
(2008)

Lançamento da
AppStore

iOS 3 (2009)

Copy & Paste;
Spotlight;
In-App Purchases.

iOS 4 (2010)

Multitasking;
Suporte a Retina Display.

iOS 8 (2014)

Extensões do App.

Curiosidade: o iOS 7 teve
grande influência na tendência
da indústria de software em
adotar um design mais limpo e
plano.

iOS 9 (2015)

Multitarefa no i

Novas fontes do sistema;

Buscas nos Apps por Sp

Atualizações nos Ap
nativos.

iOS 7 (2013)

Novo design flat,
abandonando o
Esquemorismo;
Central de Controle;
Full Multitasking.

iOS 5 (2011)

Central de Notificações;
iCloud;
Integração com redes
sociais (Twitter).

iOS 6 (2012)

Apple Maps;

Diversas mudanças nos
Apps nativos.

Curiosidade: esse é considerado um dos piores releases do iOS. Removeu o Apps nativos do Google (Maps, YouTube, etc) em resposta ao Android, e mudou a estrutura de diversas partes do sistema.

VERSÕES DE iOS

iOS 10 (2016)

Remodelagem das Notificações, mais possibilidades de Interações

iOS 11 (2017)

Customização da Central de Control, ARKit (realidade aumentada) e Core ML (Machine Learning local)

iOS 12 (2018)

Traz aprimoramentos ao iOS 11 e alguns novos recursos para desenvolvedores

FRAGMENTAÇÃO

A fragmentação de versões do iOS é bem menor do que no Android.

Graças ao controle total que a Apple tem sobre os dispositivos disponíveis para a plataforma no mercado.

FRAGMENTAÇÃO

O ciclo de releases, desde a versão 4.0, tornou-se previsível

- 1 Major release por ano

- A nova versão é apresentada no WWDC (conferência anual da Apple para desenvolvedores)

- Nele a Apple apresenta todas as novidades da nova versão para os desenvolvedores

- Um período de Beta testes acontece durante 3 a 4 meses (geralmente entre junho e setembro)

- Até 4 minor releases até fechar o ciclo da versão

FRAGMENTAÇÃO

Geralmente cada nova versão deixa de suportar dispositivos mais antigos, sendo que alguns dispositivos tiveram ciclos de atualização maiores.

Excepcionalmente o iOS 12 irá manter compatibilidade com todos os dispositivos suportados pela versão 11.

DISPOSITIVOS



iPhone

Principal dispositivo suportado pelo iOS

Lança as novas tecnologias que serão depois adaptadas a outros devices...

Multi-Touch Capacitivo

Telas Retina

Touch ID

3D Touch

Câmera Dupla com Zoom Óptico

DISPOSITIVOS

iPad

Tablets da Apple

Atualmente em 4 tamanhos:

Mini (7,9')

Air e New iPad (9,7')

Pro (10,5' e 12,9')

Processadores mais poderosos e
mais memória que os iPhones.

Suportam algumas funções
específicas como Split-View e PIP

Acessórios específicos como Apple
Pen



DISPOSITIVOS



iPod Touch

Um iPhone sem a função de telefone.

Ciclo de releases mais lento.

Tende a ser descontinuado pela Apple.

Comumente usado como dispositivo de testes por ser mais 'barato'.

DISPOSITIVOS

Apple TV

Dispositivo para levar funções interativas e de mídia as TV's.

Até a 3^a geração usava uma versão modificada do iOS.

Interface otimizada para TV

Não tinha loja de Aplicativos

A partir de 4^a geração foi redesenhada e introduziu o TvOS.

Pode projetar a tela de um iPhone ou iPad através do AirPlay.



DISPOSITIVOS



Apple Watch

Smartwatch da Apple

Utiliza o WatchOS que é derivado do iOS

Depende de um iPhone para funcionar

Seus Apps interagem diretamente com o iPhone em que esta sincronizado



ARQUITETURA E RECURSOS

ARQUITETURA

Derivado do OSX

O iOS foi adaptado a partir do código do OSX

Portanto também da família UNIX

Utiliza o núcleo Darwin/BSD

O iOS é otimizado para a arquitetura de processadores ARM

ARQUITETURA

Apps executam no modelo 'Sandbox'

O App executa num ambiente controlado, uma 'caixa de areia'

Os Apps não tem conhecimento e não podem acessar os processos de outros Apps ou se comunicar diretamente com eles

Inibe a ação de Apps nocivos já que eles não podem prejudicar outras partes do sistema

ARQUITETURA

Proporciona serviços para acessar recursos e sensores presentes nos dispositivos

Serviço de Localização (GPS)

Acelerometro/Giroscópio

Camera

Microfone

Implementa um modelo de segurança e privacidade baseado em permissões

ARQUITETURA

Implementa um modelo de segurança e privacidade baseado em permissões

Modelo rígido de distribuição de Apps

Somente Apps assinados com certificados digitais válidos podem ser instalados

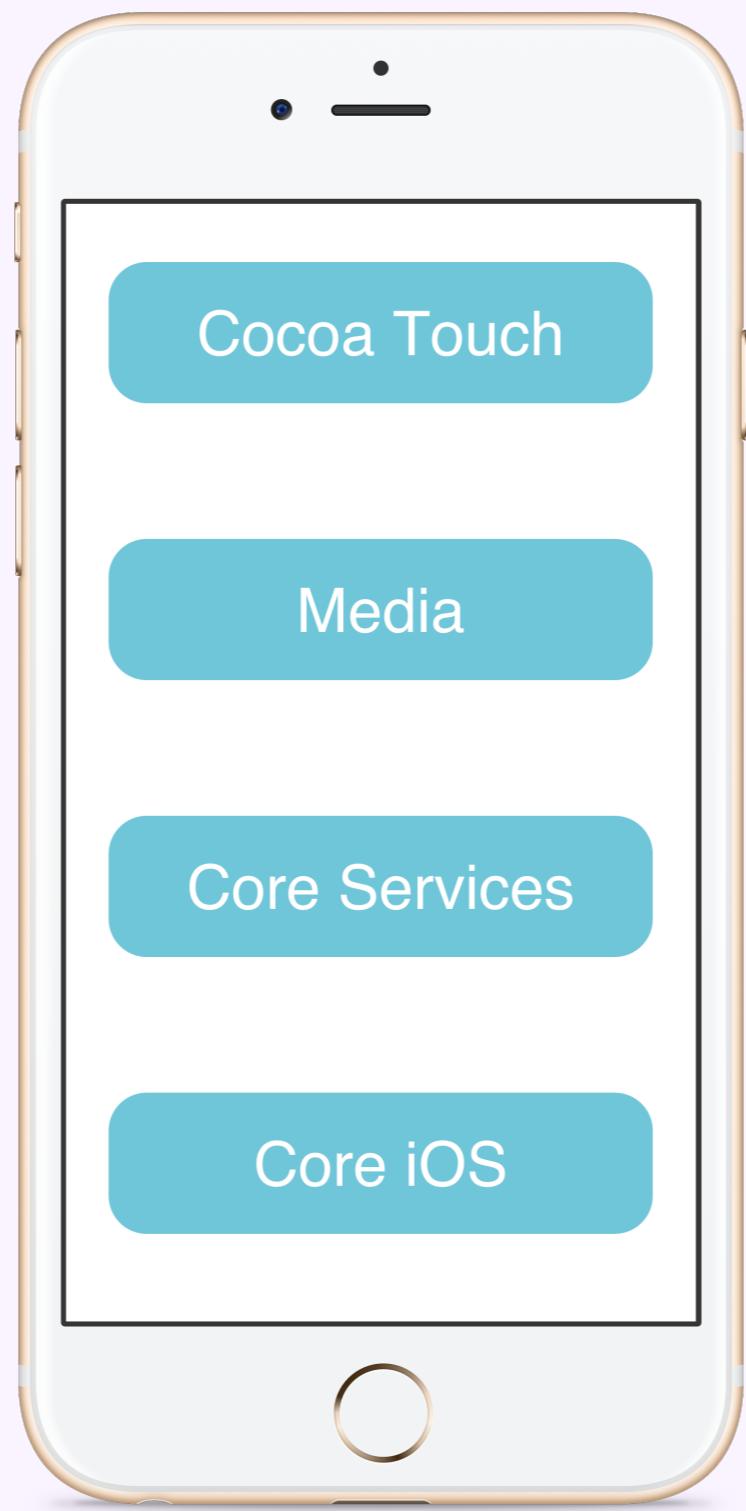
Usuários comuns somente podem instalar seus Apps a partir da App Store

RECURSOS DA PLATAFORMA

A plataforma iOS é dividida em diversas camadas oferecendo os recursos necessários para desenvolvimento de Apps.

As camadas mais baixas estão mais próximas do hardware enquanto as superiores implementam níveis de abstração que facilitam a implementação de funcionalidades comuns.

PRINCIPAIS CAMADAS



CORE iOS

Acessam os componentes e serviços fundamentais do Sistema Operacional.

Componentes

Kernel do OSX

Gerenciamento de Energia (Power Management)

Mach 3.0

Keychain Access

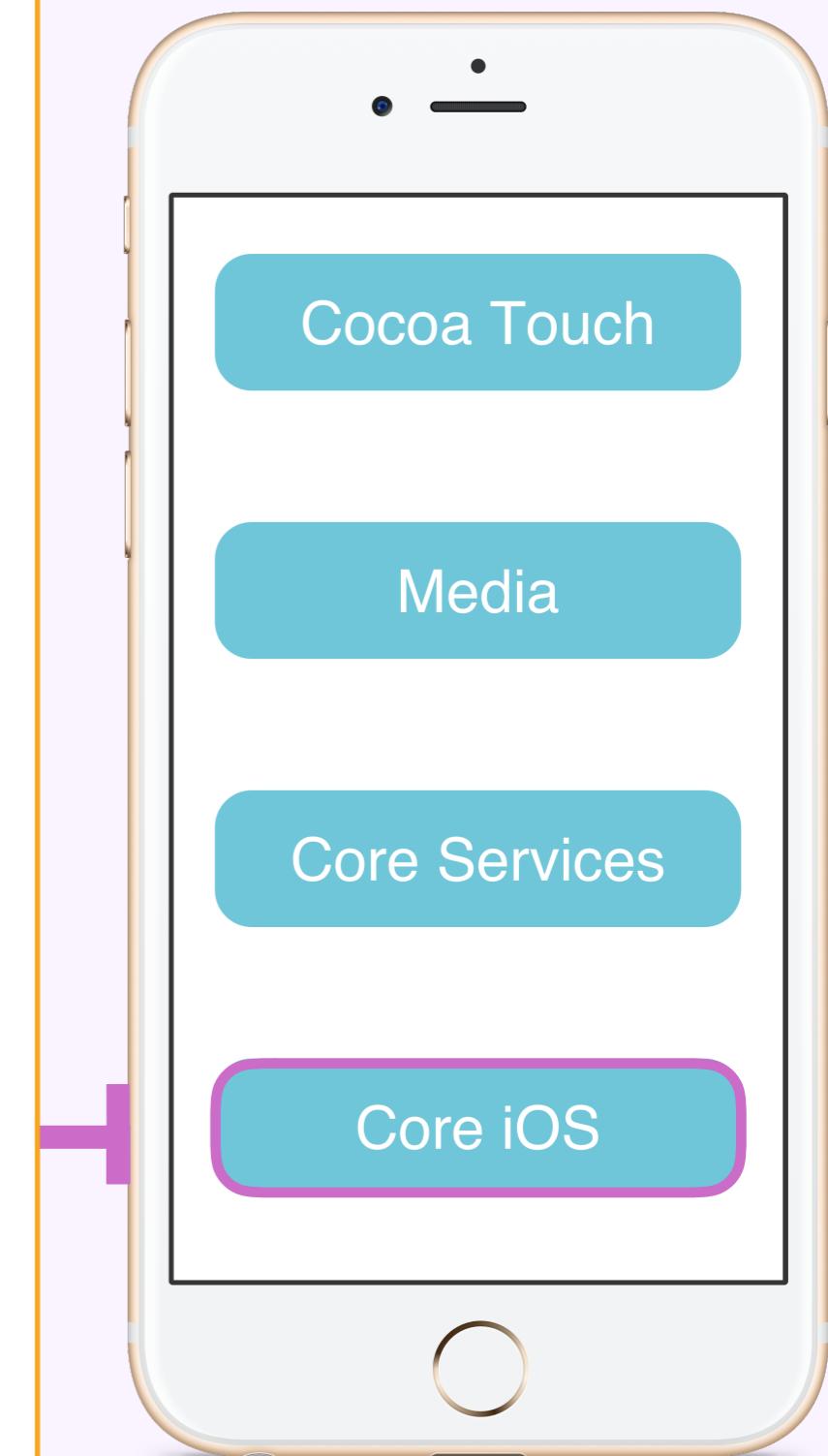
Certificados

Sockets

Sistema de arquivos

Segurança

Bonjour



CORE SERVICES

Bibliotecas e serviços

Componentes

Coleções (Collection)

Localização Geográfica (Core Location)

Contatos (Address Book)

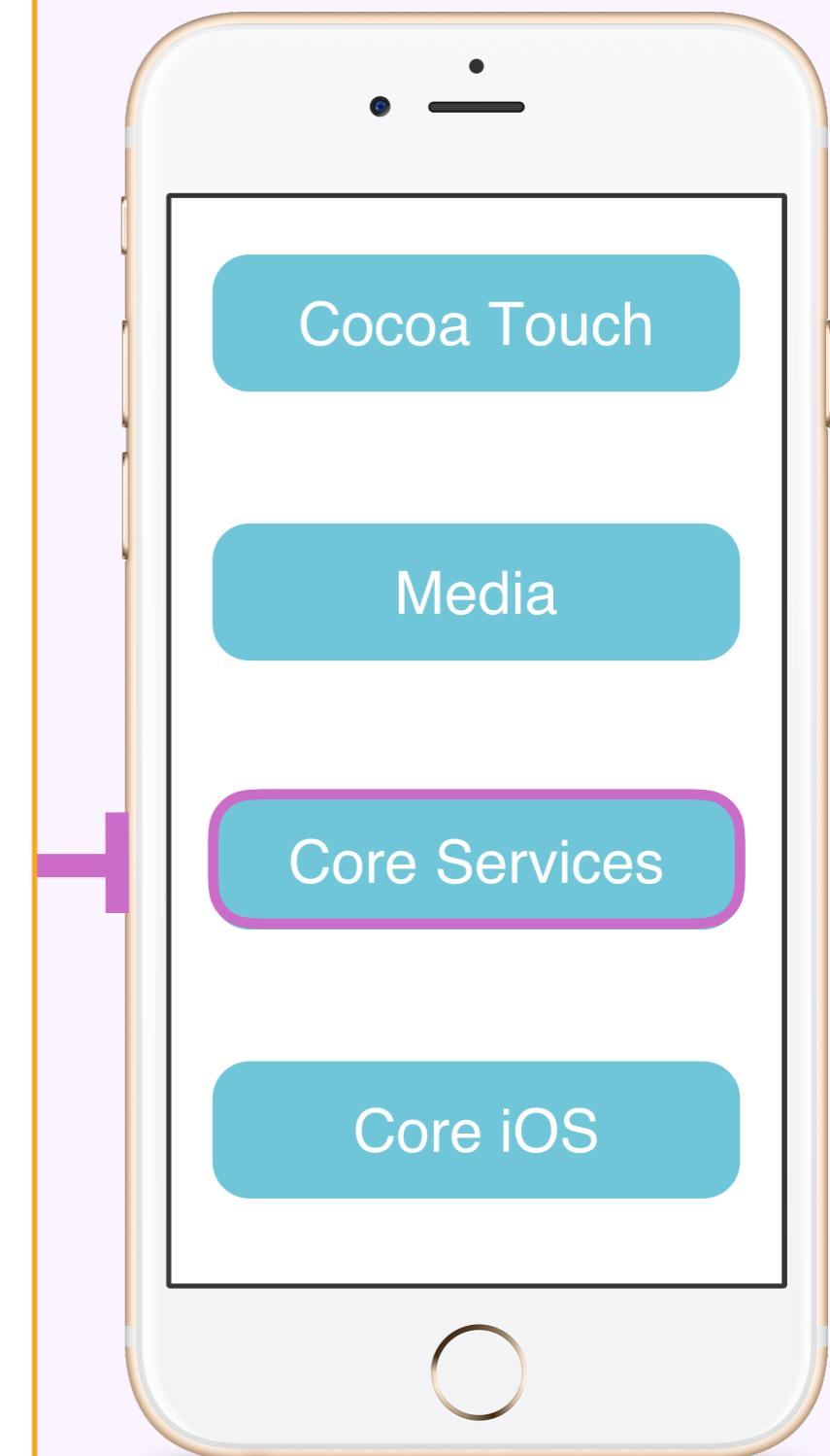
Serviços de rede

Threading

Acesso a arquivos

Preferências

SQLite



MEDIA

Bibliotecas e serviços para reprodução e manipulação de mídia

Componentes

Reprodução de Áudio (Core Audio)

OpenAL

JPEG, PNG, TIFF

PDF

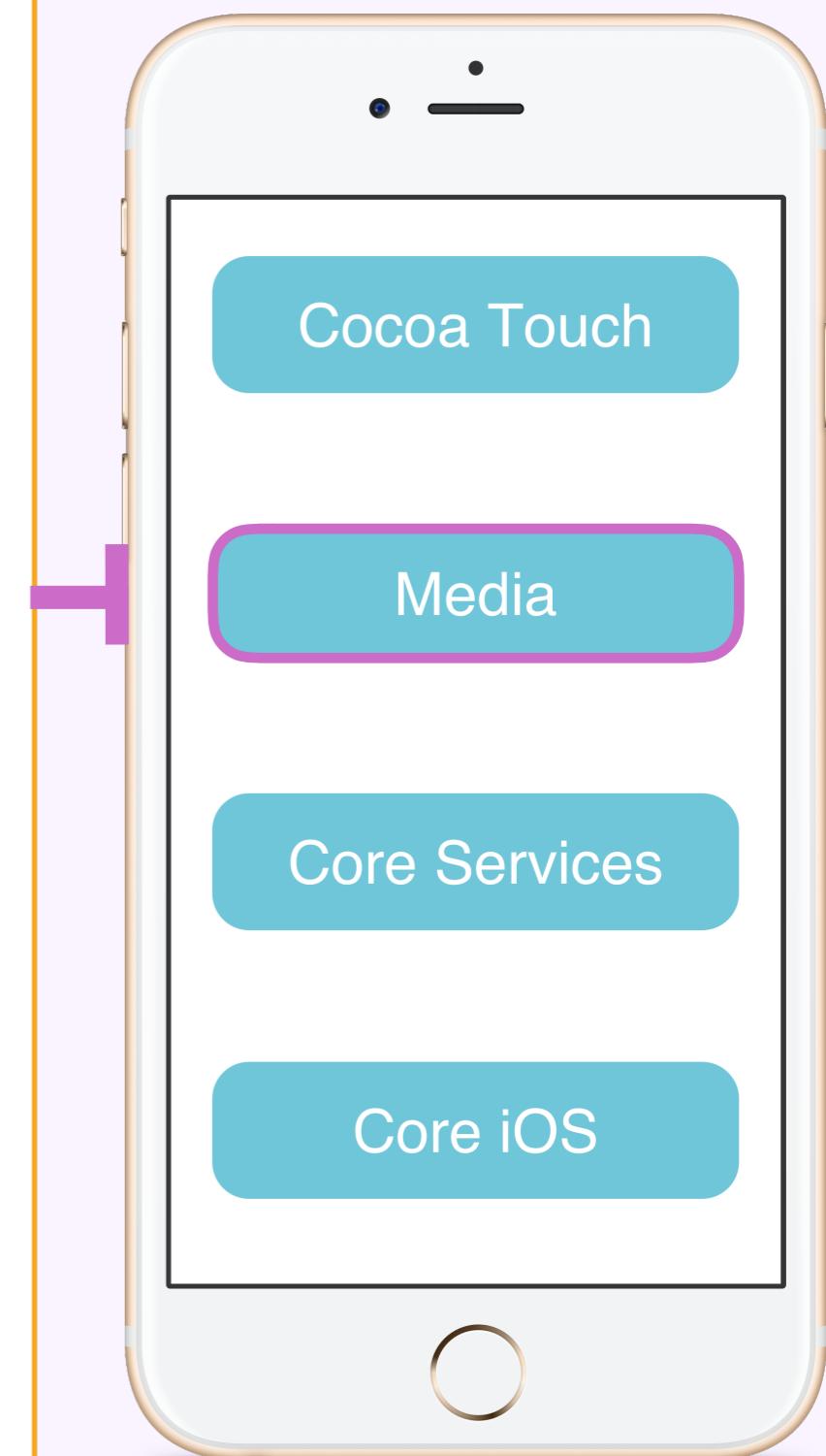
Gravação/Mixagem de Áudio

Desenhos e animações 2D (Quartz)

Animação (Core Animation)

Reprodução de Vídeo

OpenGL ES

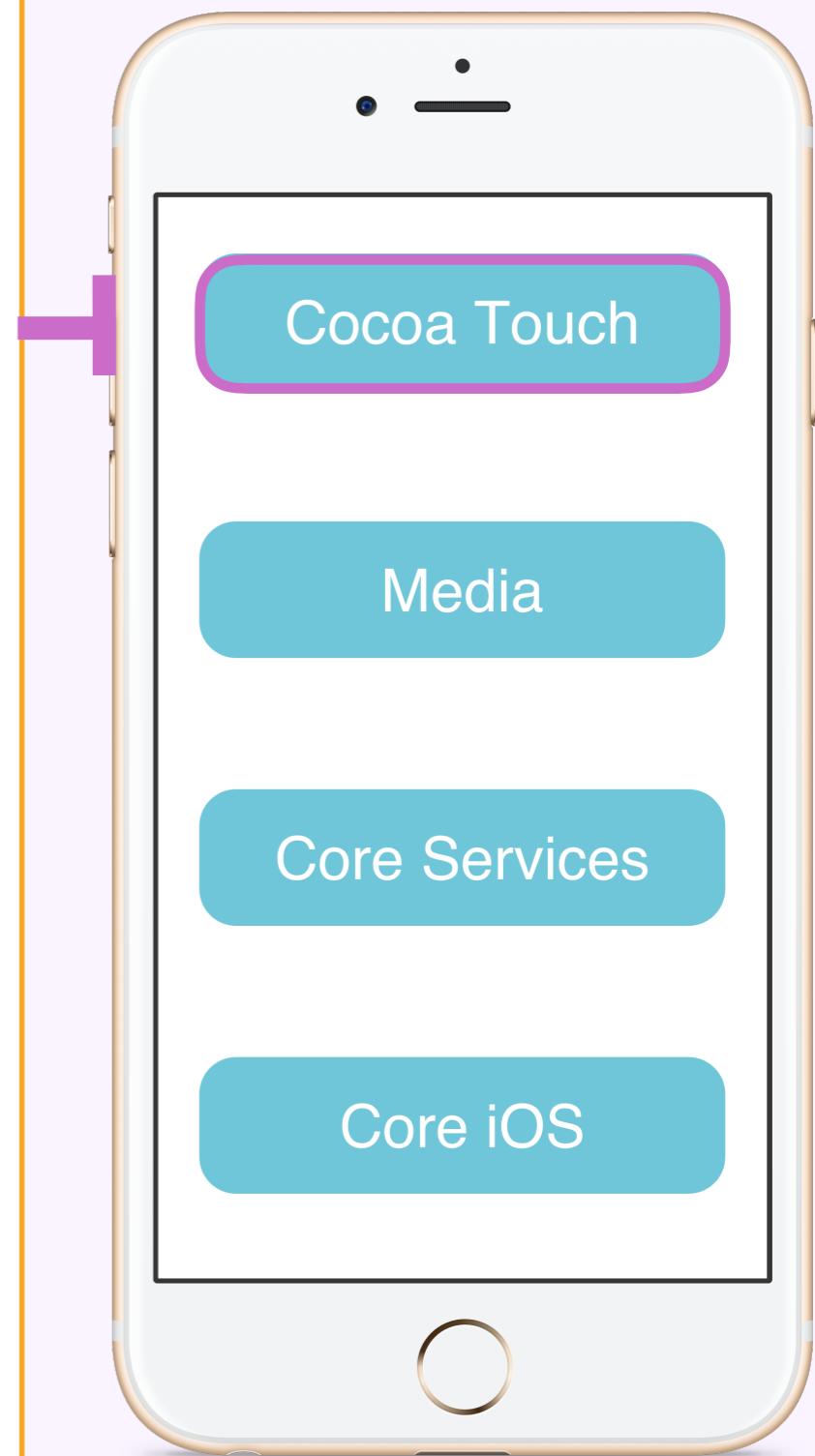


COCOA TOUCH

Bibliotecas de mais alto nível na plataforma do iOS.

Possibilita a criação de interfaces, processamento de entrada e acesso alto nível aos sensores e serviços do sistema.

Observação: nosso treinamento é focado nessa camada



COCOA TOUCH

Componentes

Views e View Controllers

Controles

Reconhecedores de Gestos (Gesture Recognizers)

Alertas

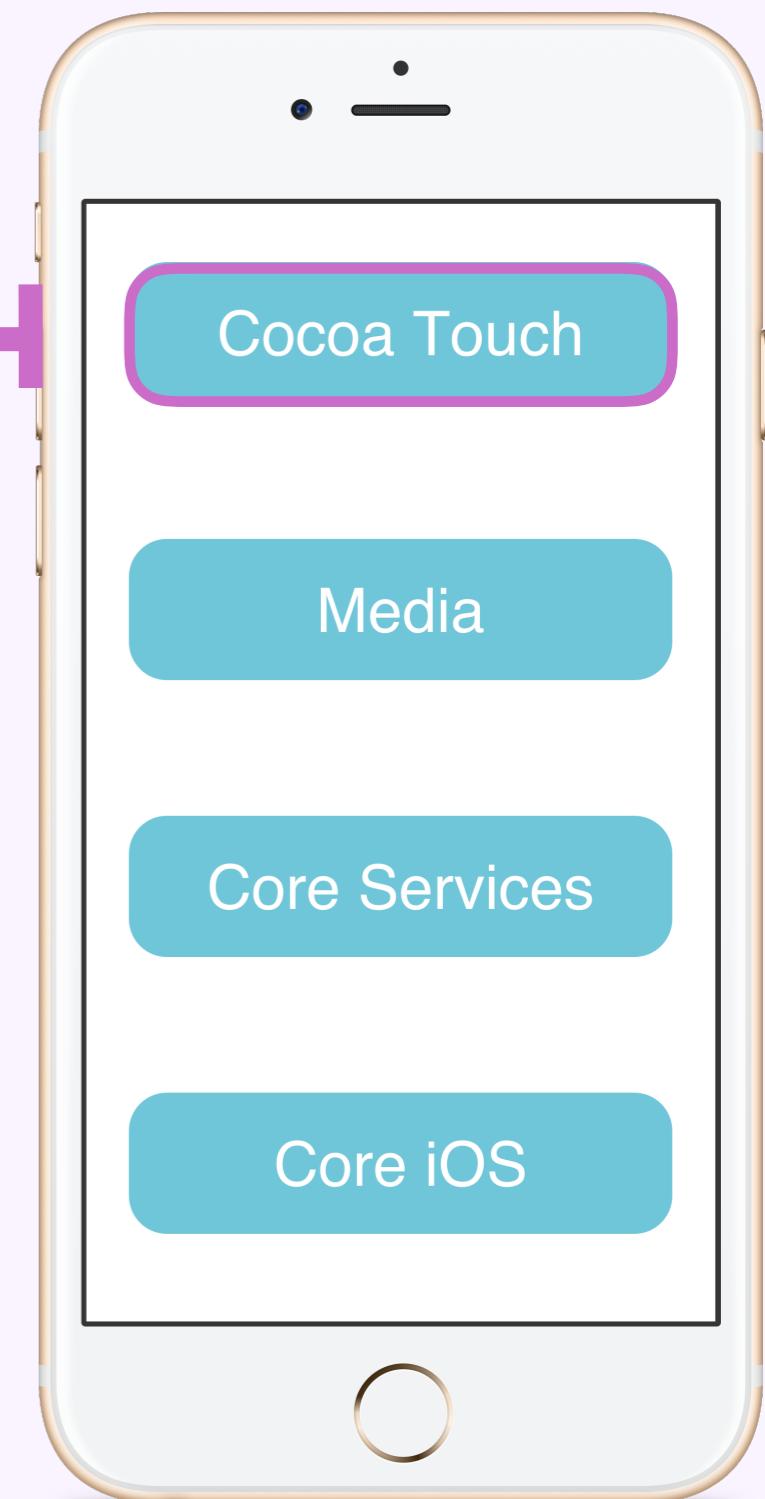
Localização e Internacionalização

MapKit

WebView

Image Picker

Observação: nosso treinamento
é focado nessa camada





COMPONENTES DA PLATAFORMA

COMPONENTES DA PLATAFORMA

Ferramentas



Xcode

Instruments

Oferecidas gratuitamente pela Apple, porém disponíveis apenas para Mac

Linguagens

Objective-C



Swift

Frameworks (Bibliotecas)

Core Data

UIKit

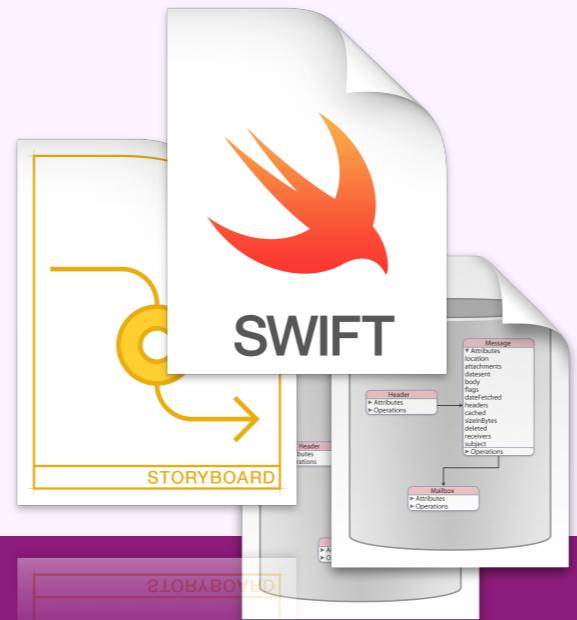
MapKit

Foundation

...

Estratégias de Design

MVC



O PATTERN MVC

O PATTERN MVC

Pattern fundamental na modelagem de aplicações
Cocoa e Cocoa Touch

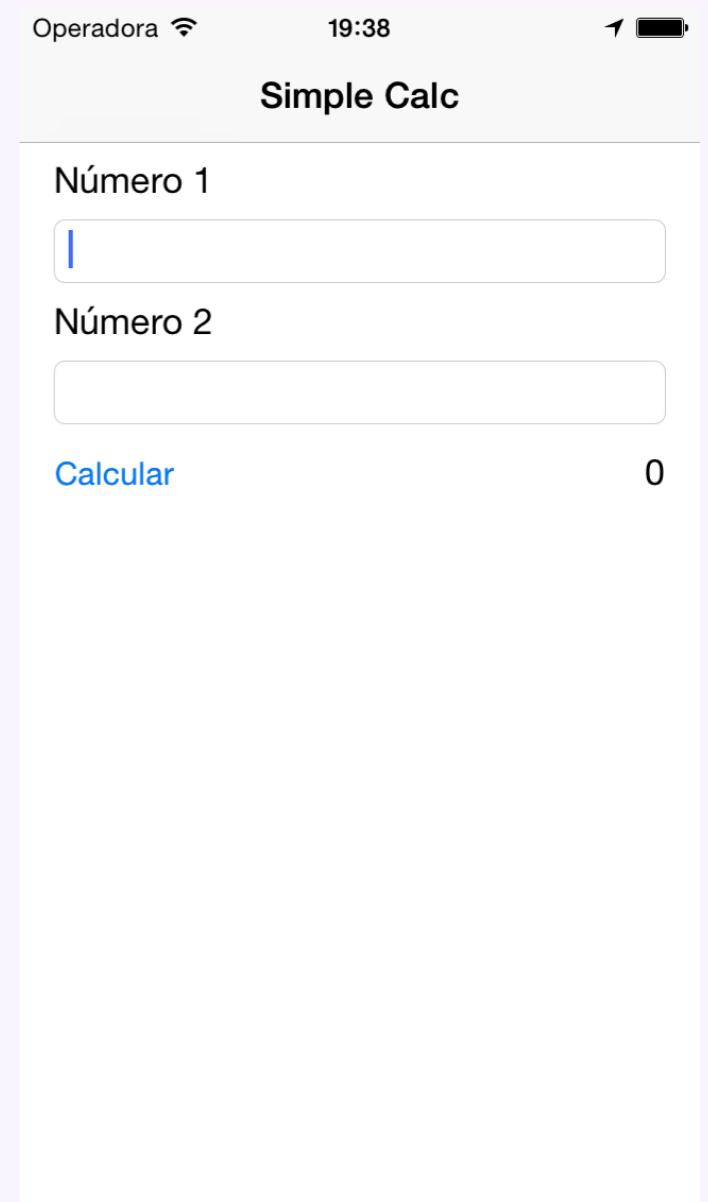
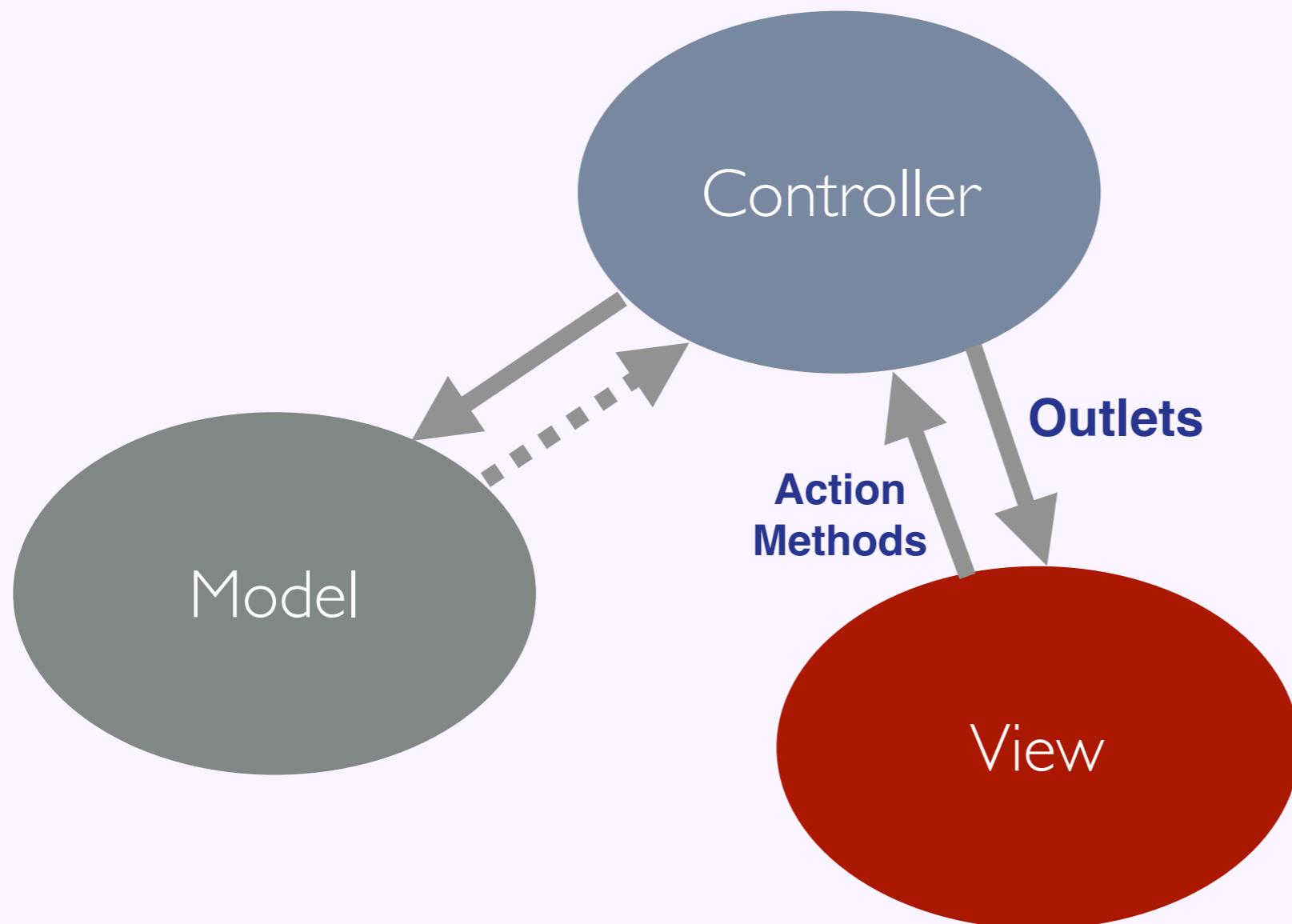
Divide a construção de uma aplicação em 3 partes fundamentais:

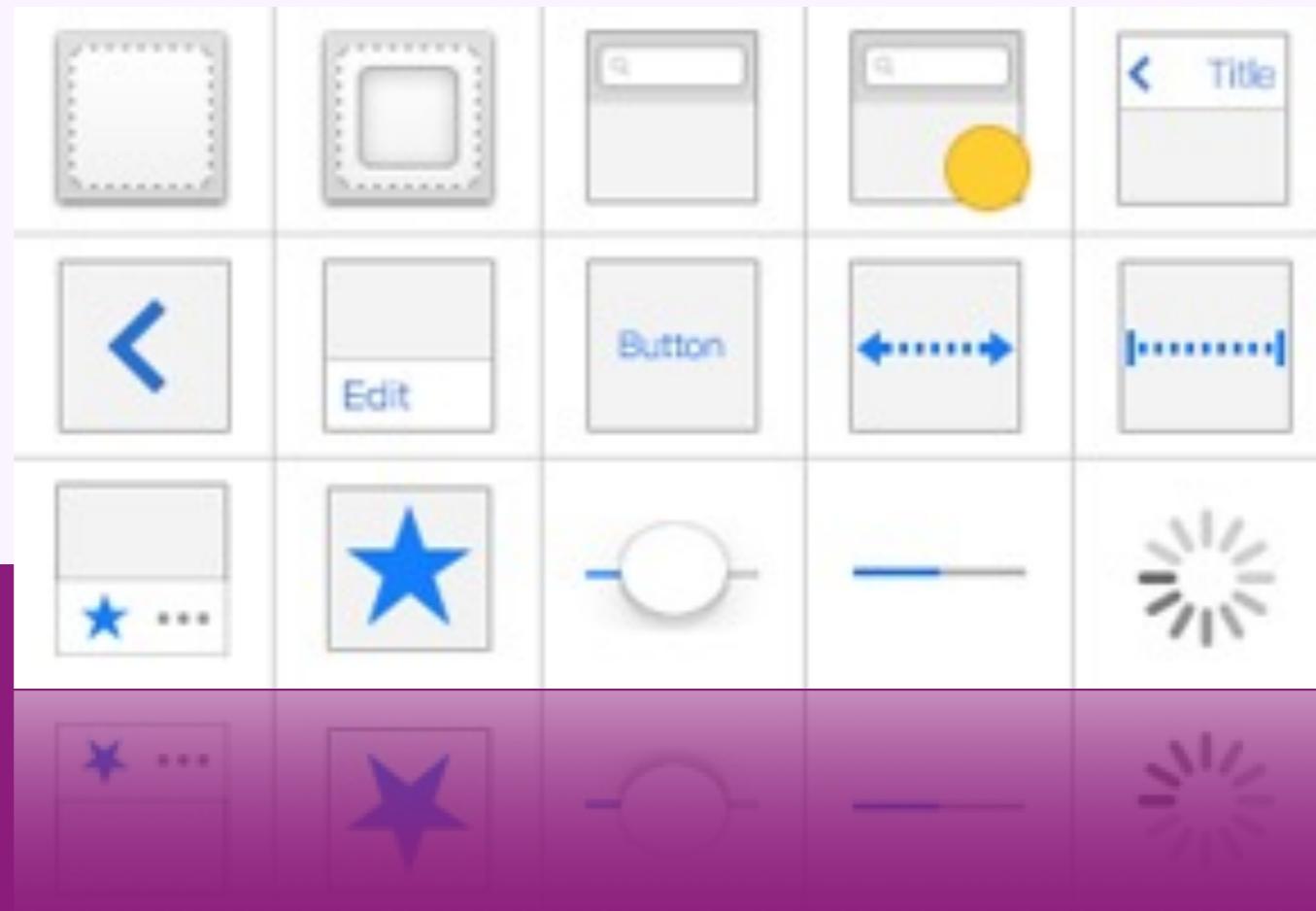
Model: responsável por toda a parte lógica e estrutural da aplicação, o seu “cérebro”.

View: responsável por apresentar a interface da aplicação, da maneira mais adequada a sua proposta.

Controller: responsável por gerenciar a apresentação da interface da aplicação e sua interação com o modelo.

O PATTERN MVC



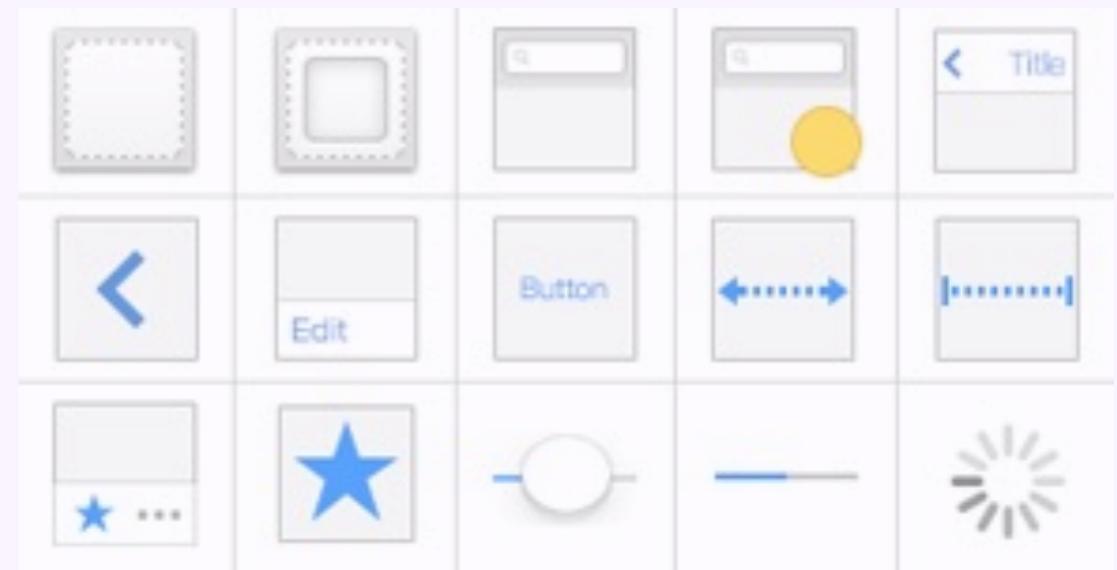


APRESENTANDO O UIKIT

APRESENTANDO O UIKIT

Framework fundamental
para a construção e
gerenciamento de Apps de
iOS.

Proporciona a arquitetura
para construção da
interface de usuário do
App, bem como suas
interações.

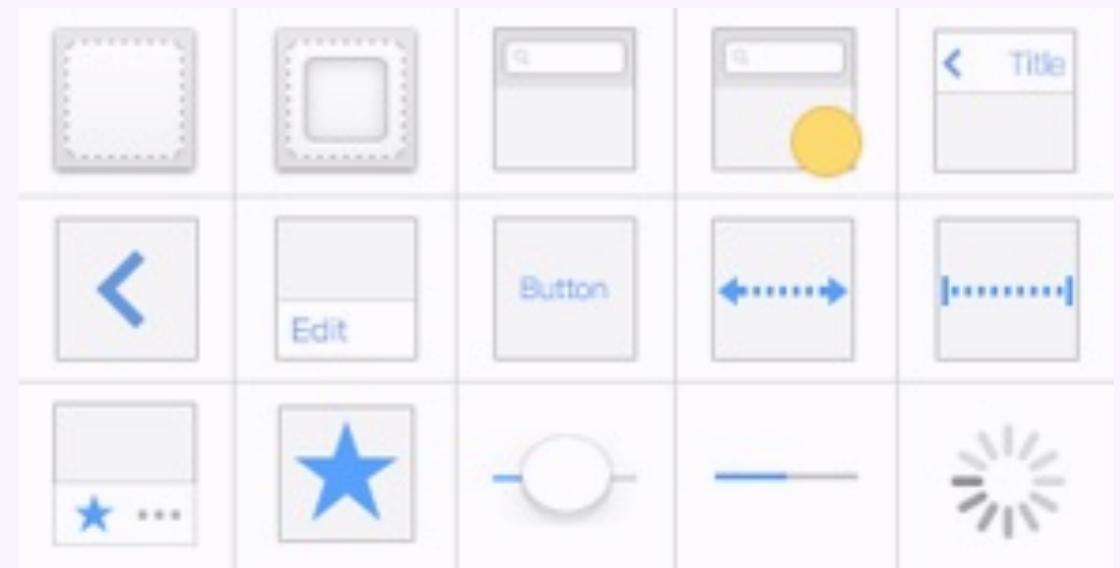


PRINCIPAIS FUNCIONALIDADES

Gerenciamento da execução
do App no plano principal ou
no plano de fundo.

Modelo de *View Controllers*
para gerenciamento de
interface.

Conjunto rico de
componentes visuais
personalizáveis.

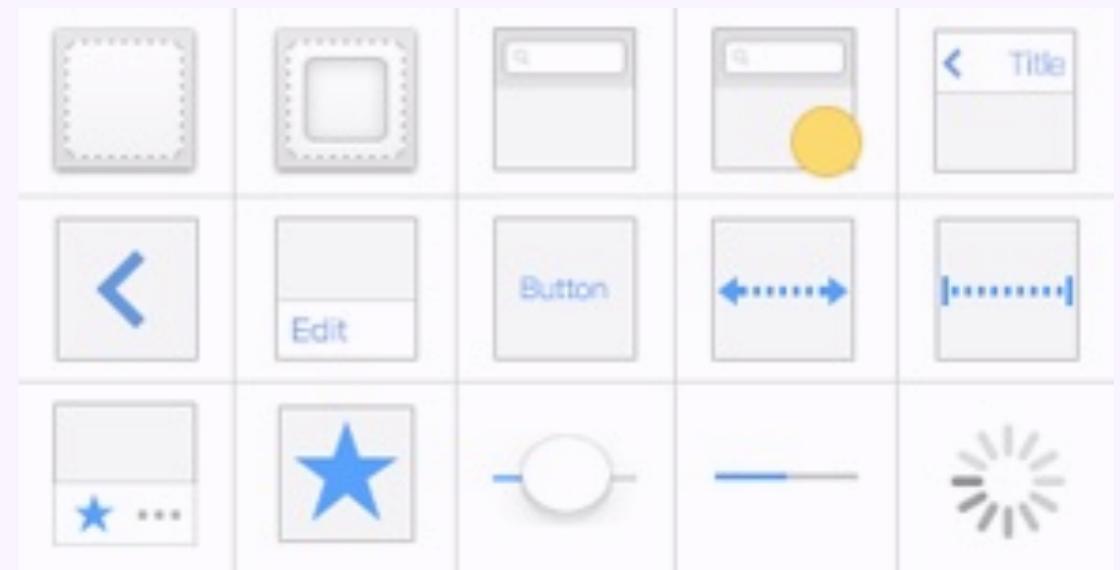


PRINCIPAIS FUNCIONALIDADES

Animações dos elementos visuais. Manipulação e tratamento de toques e gestos.

Pasteboard - operações de copiar, cortar e colar.

Acesso às câmeras e à biblioteca de imagens do usuário.





CLASSES DO UIKIT

CLASSES DO UIKIT

