



Arquitetura - IOS Apps

Autor: Daniela Mazzi

O que é esperado de um app?

- * Boa experiência de usuário
- * Bom design e interface
- * Rápido
- * Responsivo
- * Utilizar mínimo de energia necessária
- * Suporte para ultimas versões dos aparelhos
IOS

O que um app deve ter?

- *Info.plist (Information Property List File) - arquivo que fornece informações críticas sobre a configuração do app
- *Declaração das capacidades requeridas pelo app (Ex: uso de microfone, camera, gps)
- *Um ou mais ícones (home)
- *Uma ou mais imagens iniciais (launch images)

Info.plist

Key	Type	Value
▼ Information Property List	Dictionary	(18 items)
Localization native development region	String	\$(DEVELOPMENT_LANGUAGE)
Executable file	String	\$(EXECUTABLE_NAME)
Bundle identifier	String	\$(PRODUCT_BUNDLE_IDENTIFIER)
InfoDictionary version	String	6.0
Bundle name	String	\$(PRODUCT_NAME)
Bundle OS Type code	String	APPL
Bundle versions string, short	String	1.0
▼ URL types	Array	(1 item)
▶ Item 0	Dictionary	(1 item)
Bundle version	String	1
FacebookAppID	String	188103488618873
FacebookDisplayName	String	LoginTeste
▼ LSAApplicationQueriesSchemes	Array	(4 items)
Item 0	String	fbapi
Item 1	String	fb-messenger-share-api
Item 2	String	fbauth2
Item 3	String	fbshareextension
Application requires iPhone environment	Boolean	YES
Launch screen interface file base name	String	LaunchScreen
Main storyboard file base name	String	Main
▼ Required device capabilities	Array	(1 item)
Item 0	String	armv7
▼ Supported interface orientations	Array	(3 items)
Item 0	String	Portrait (bottom home button)
Item 1	String	Landscape (left home button)
Item 2	String	Landscape (right home button)
▼ Supported interface orientations (iPad)	Array	(4 items)
Item 0	String	Portrait (bottom home button)
Item 1	String	Portrait (top home button)
Item 2	String	Landscape (left home button)
Item 3	String	Landscape (right home button)

Privacidade de Usuário - Boas Práticas



- *Revisar guias governamentais e industriais de privacidade mobile
- *Pedir acesso a dados sensíveis do usuário ou aparelho quando necessário
- *Transparência com usuário sobre o uso de seus dados (política de privacidade e resumo na descrição do app)
- *Liberdade para o bloqueio do acesso a tipos de dados
- *Utilizar mínimo de dados do usuário necessários
- *Proteger dados coletados do usuário (IOS data protection e App Transport Security)
- *Não utilizar identificador único do aparelho (UDID). Utilizar identifierForVendor (UIDevice) ou advertisingIdentifier (ASIdentifierManager)
- *Aplicativos que suportam entrada de audio devem apenas iniciar a gravação no ponto desejado

Internacionalização do aplicativo

- *Aplicativos devem ser internacionalizados para um alcance maior de usuários
- *Para isso criam se grupos de arquivos com linguagens especificas para adicionar ao projeto
- *Um aplicativo típico de IOS requiere versões localizadas dos seguintes arquivos:
 - *Arquivos de storyboard (atualizar campos de texto e layout)
 - *Arquivos de string
 - *Arquivos de imagens (evitar imagens com texto)
 - *Vídeos e arquivos de áudio

Ciclo de Vida do Aplicativo

- *Frameworks em IOS contam com padrões de design como MVC e Delegation
- *Função Main: ponto de entrada, passa controle para o UIKit framework chamando a função UIApplicationMain
- *UIApplicationMain cria os objetos principais do aplicativo, carrega a interface de usuário, executa código customizado para configurações iniciais e executa o aplicativo.
- *UIApplication object: gerencia loop de eventos e outros comportamentos de alto nível do app. Também reporta transições chaves do app e outros eventos especiais, como push notifications.
- *App delegate object: em conjunto com o UIApplication lida com a inicialização do app, mudanças de estado e eventos de alto nível

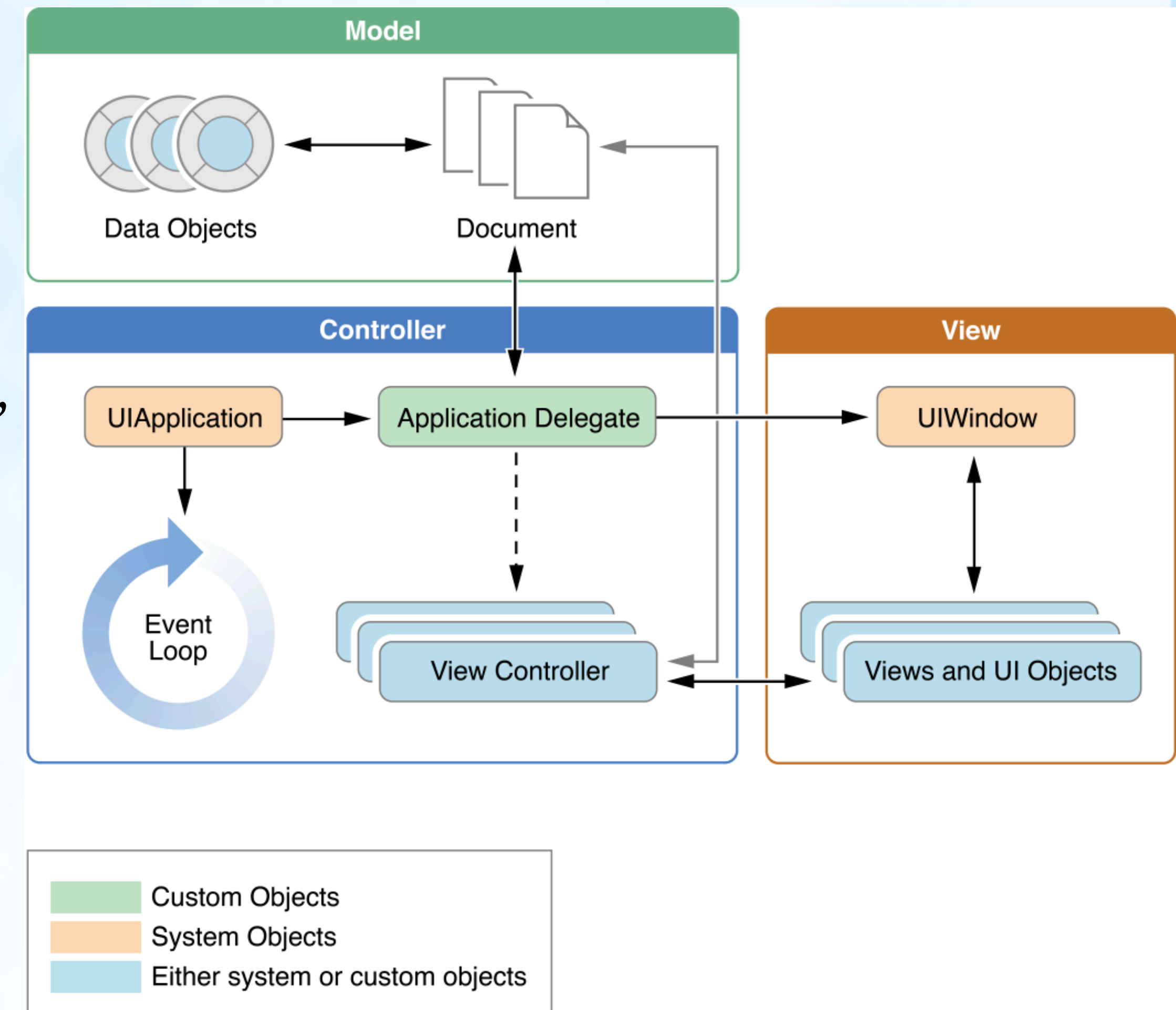
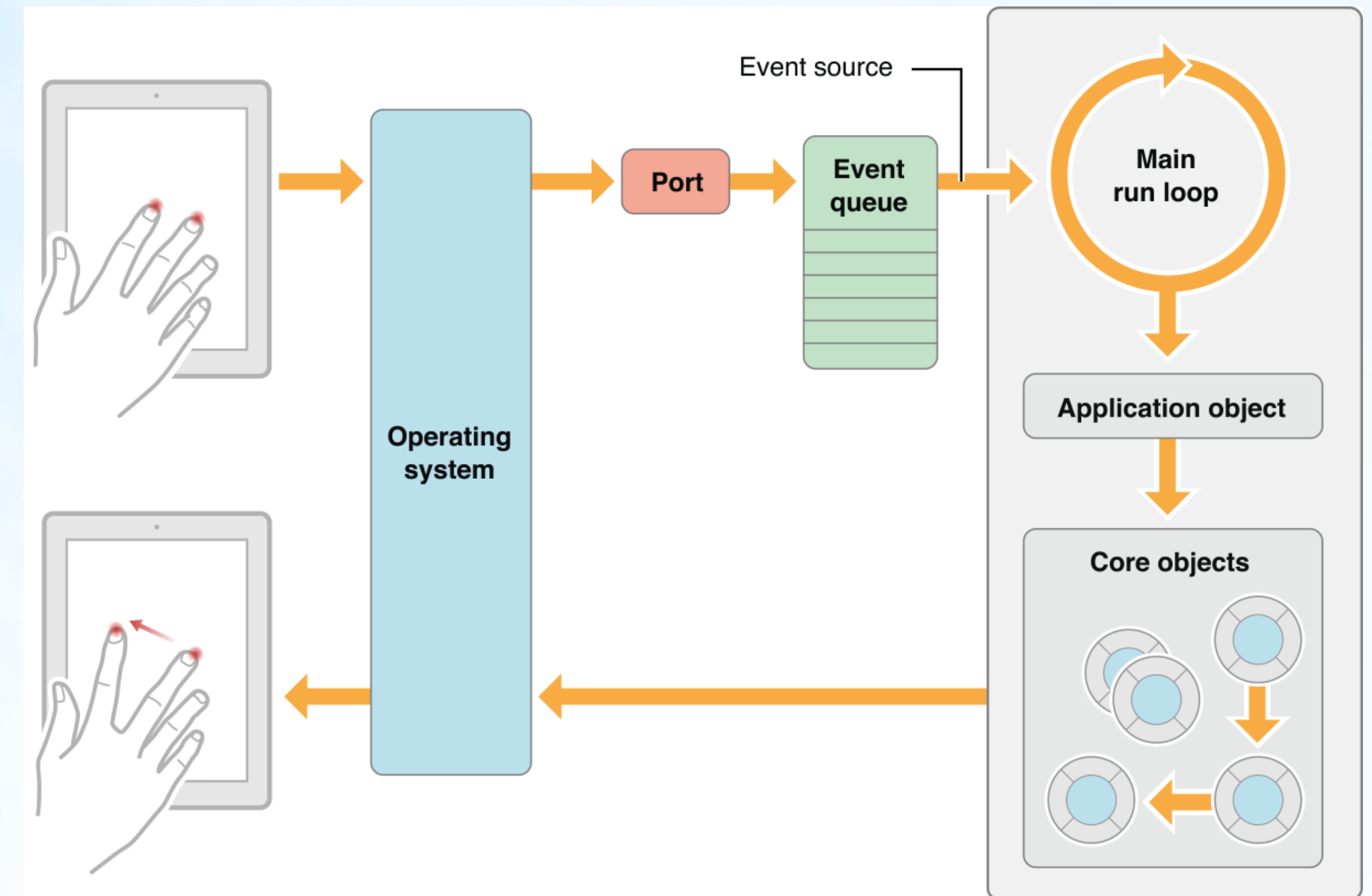


Figura - Objetos chaves de um aplicativo IOS

Ciclo de Vida do Aplicativo: Loop de execução principal

- * UIApplication configura o loop de execução principal na inicialização do aplicativo e o utiliza para processar eventos e lidar com atualizações de interfaces.
- * É executado na thread principal (main thread)
- * Onde todos os eventos relacionados ao usuário devem ser processados.



- * Main thread do aplicativo é criada pelo sistema.
- * Pode-se criar threads adicionais para suportar concorrência de tarefas
- * Grand Central Dispatch (GCD)
- * UI activities devem acontecer na MainQueue
- * Tarefas de longa duração devem ser executadas em uma thread de background
- * Ao iniciar o app, deve-se deixar apenas tarefas que contribuam para a inicialização da interface de usuário na thread principal.

*Exemplo de método de busca de imagem a partir de uma url passada (imageUrl) utilizando threads:

```
private func fetchImage() {  
    if let url = imageUrl {  
        spinner.startAnimating()  
        DispatchQueue.global(qos: .userInitiated).async { [weak self] in    //Criação de uma thread concorrente para carregar a imagem da url  
            let urlContents = try? Data(contentsOf: url)  
            DispatchQueue.main.async { //Volta-se a thread principal para exibir a imagem na interface, pois todos os elementos da interface do usuário (UI) devem acontecer nessa thread  
                if let imageData = urlContents, url == self?.imageUrl {  
                    self?.image = UIImage(data: imageData)  
                }  
            }  
        }  
    }  
}
```

*Como a imagem vem de uma url externa, ela pode demorar para ser carregada, nesse caso a tela do usuário não deve travar enquanto se espera a imagem carregar, então se cria uma thread concorrente enquanto a thread principal carrega os elementos da tela e , assim que a imagem carregar, ela é exibida.

Ciclo de Vida do Aplicativo: Estados de Execução

*Aplicativo muda de estado em estado em resposta a ações que acontecem no sistema.

*Estados do Aplicativo:

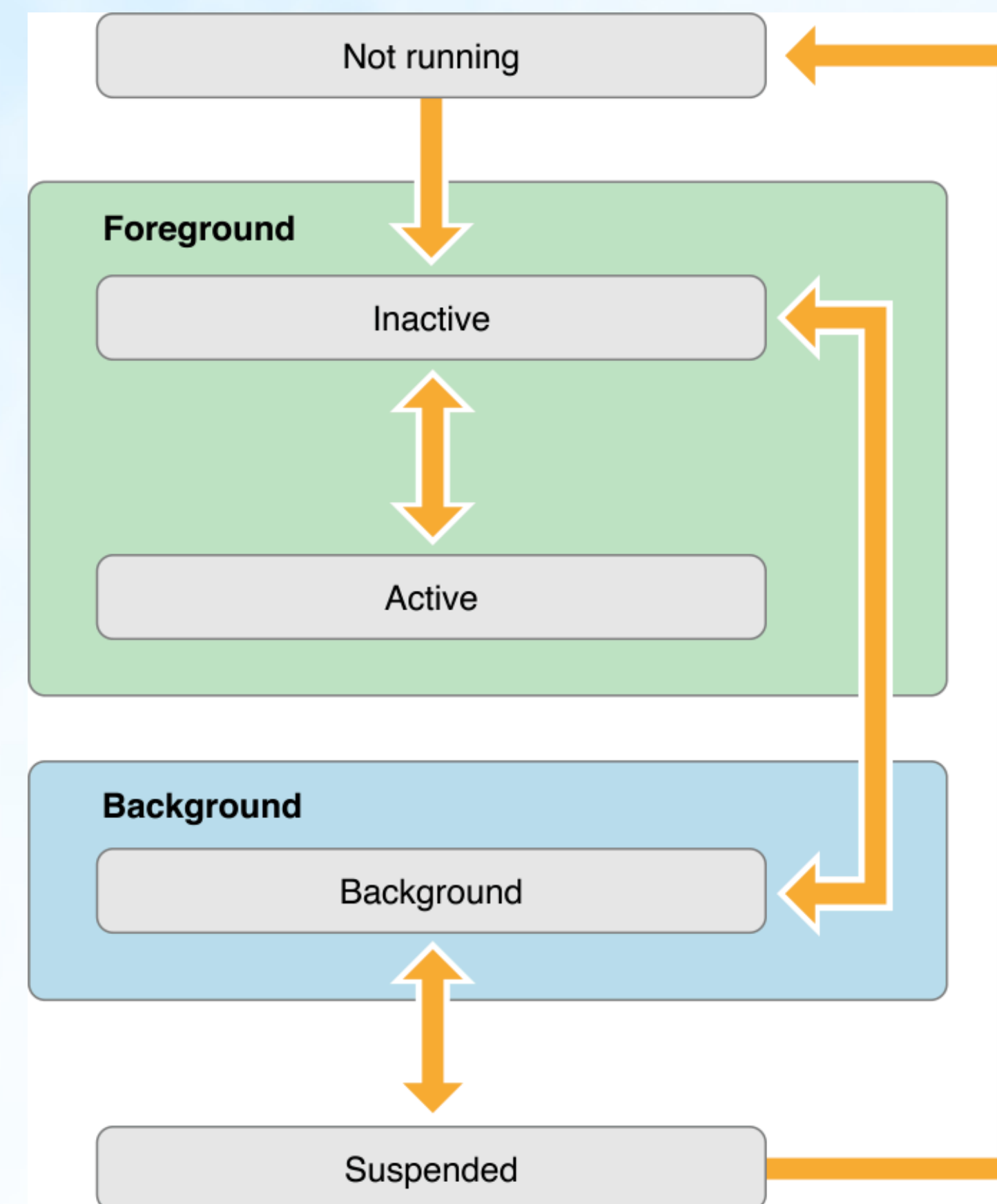
*Not running

*Inactive

*Active

*Background

*Suspended



* Métodos chamados durante transições de estados:

* **application:willFinishLaunchingWithOptions** - primeira oportunidade de executar código na inicialização (determinar se estado de restauração deve ocorrer)

* **application:didFinishLaunchingWithOptions** - usado para ajustes finais na interface de usuário

* **applicationDidBecomeActive** - app acabou de entrar em foreground (preparações de último minuto)

* **applicationWillResignActive** - app está saindo de foreground, usado para colocar app em um modo inativo.

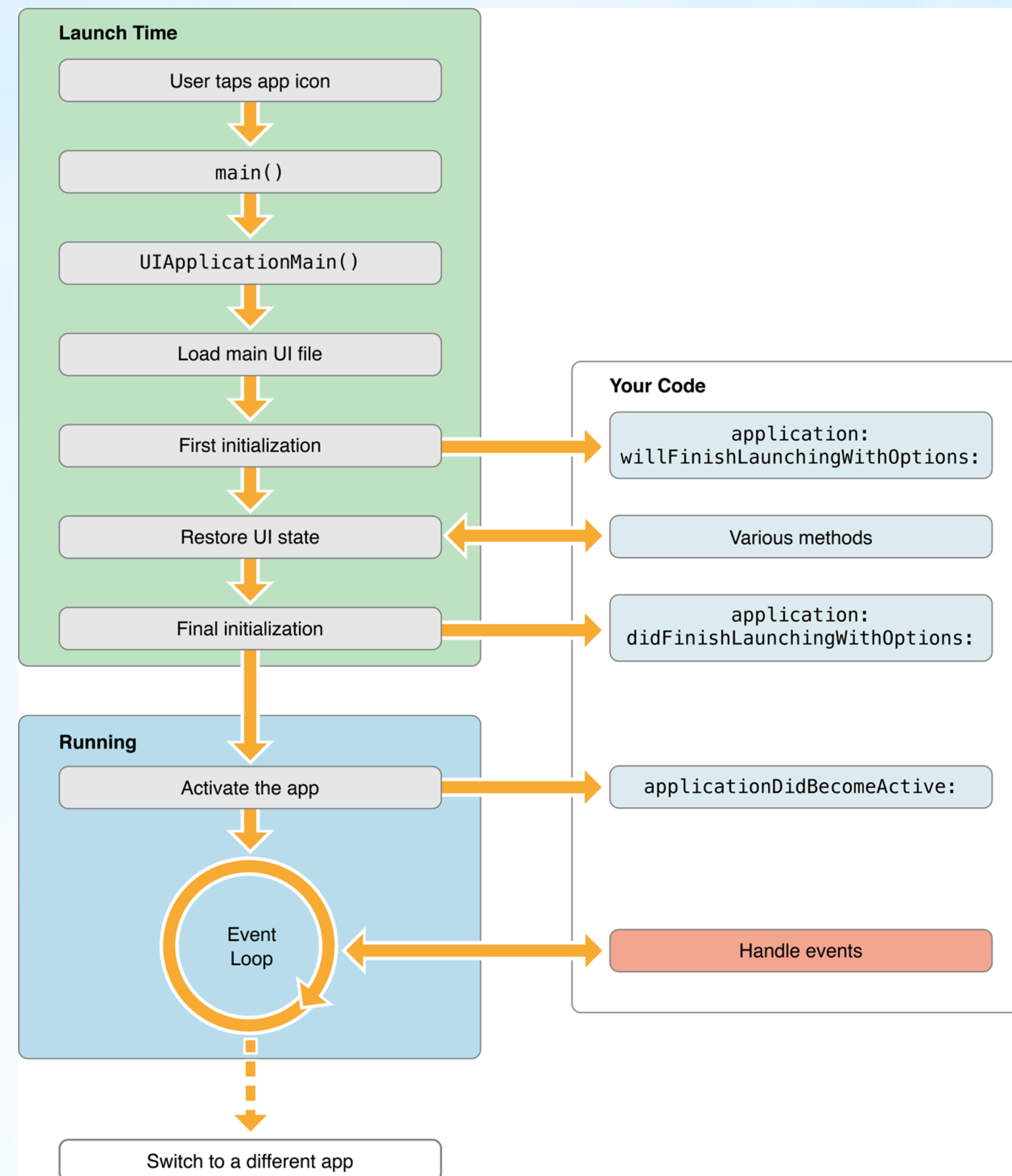
* **applicationDidEnterBackground** - app está em background e pode ser suspenso a qualquer hora

* **applicationWillEnterForeground** - app está voltando para foreground, mas ainda não está ativo

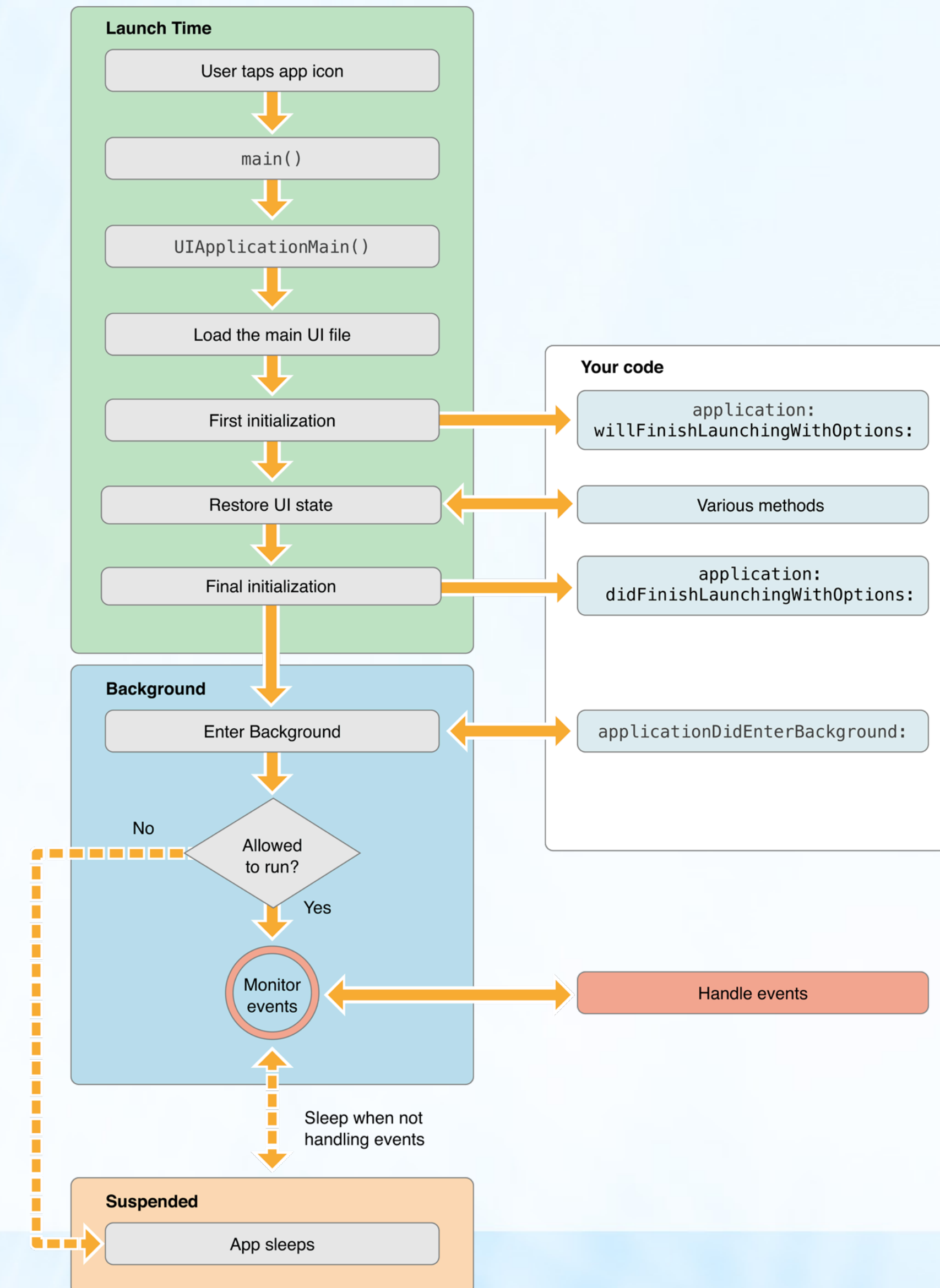
* **applicationWillTerminate** - app está sendo terminado, método não é chamado se o app está suspenso

Ciclo de inicialização do aplicativo

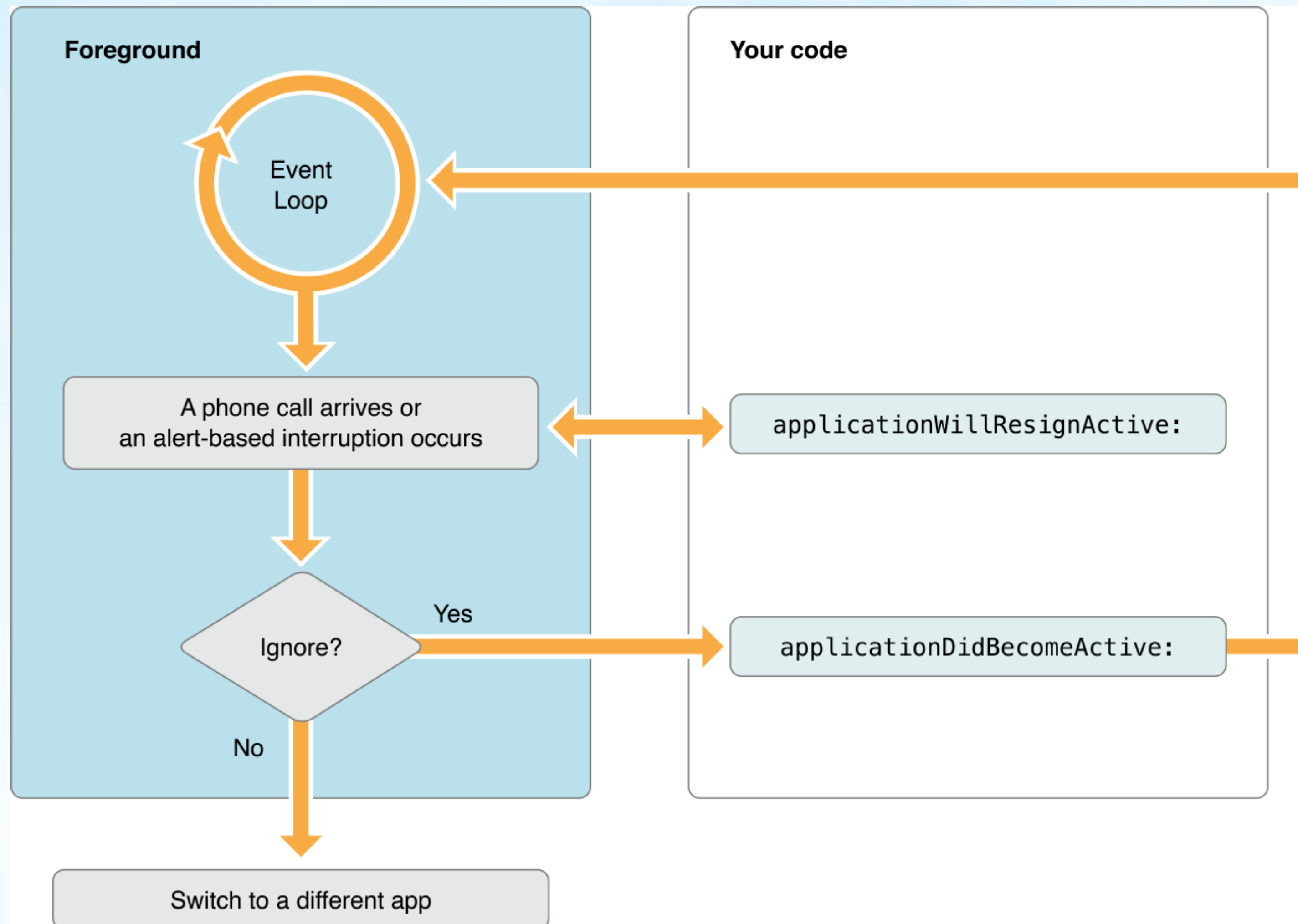
Inicialização em foreground



Inicialização em background



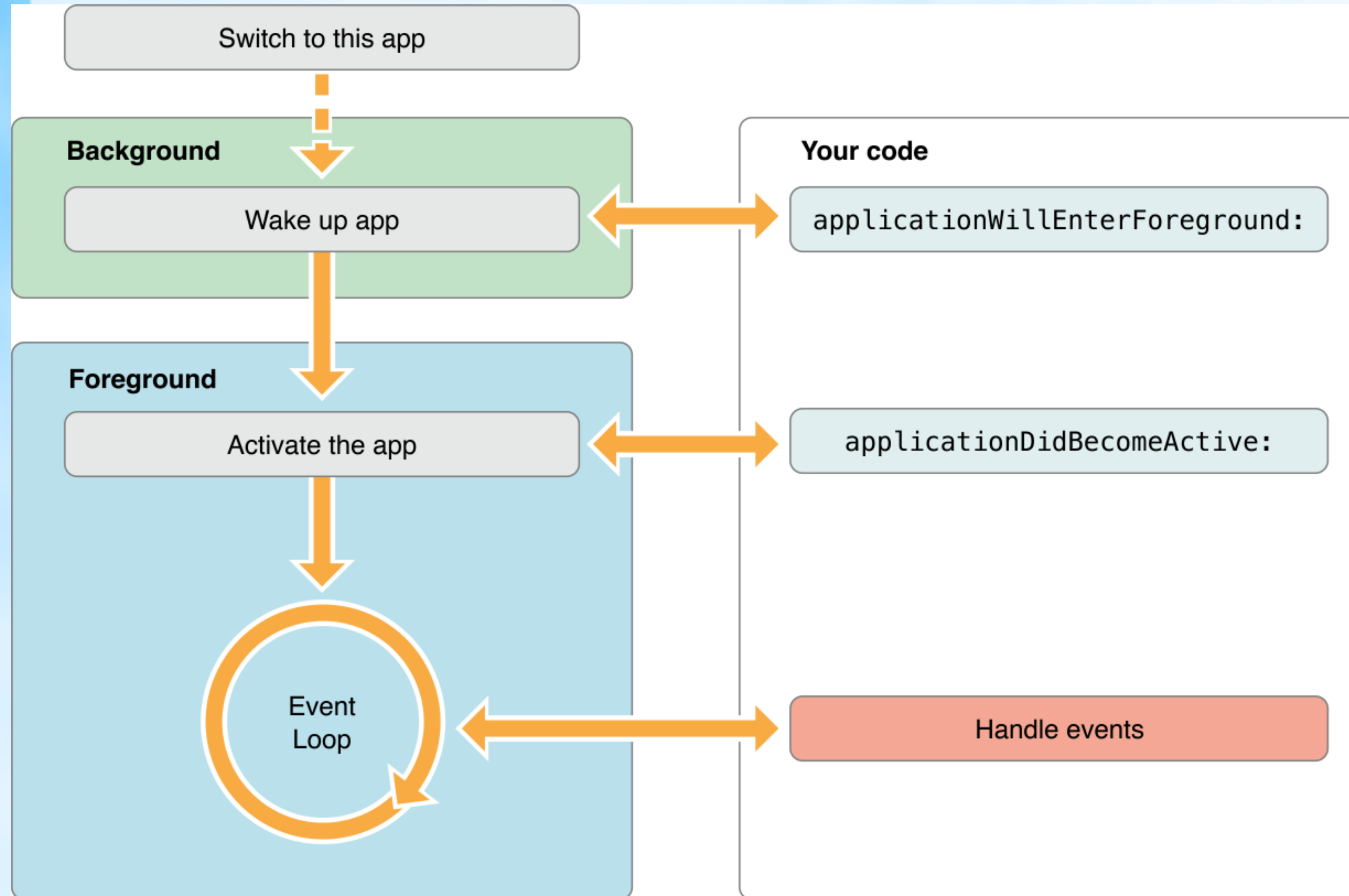
Como lidar com interrupções temporárias



applicationWillResignActive: salvar dados, parar timer e tarefas periódicas, parar de rodar queries de metadata, não iniciar novas tarefas, pausar playbacks, suspender filas executando código não crítico)

applicationDidBecomeActive: reverter o que foi feito no **applicationWillResignActive**

Transição Background - Foreground



`applicationWillEnterForeground:`
reverter o que foi feito no
`applicationDidEnterBackground`)

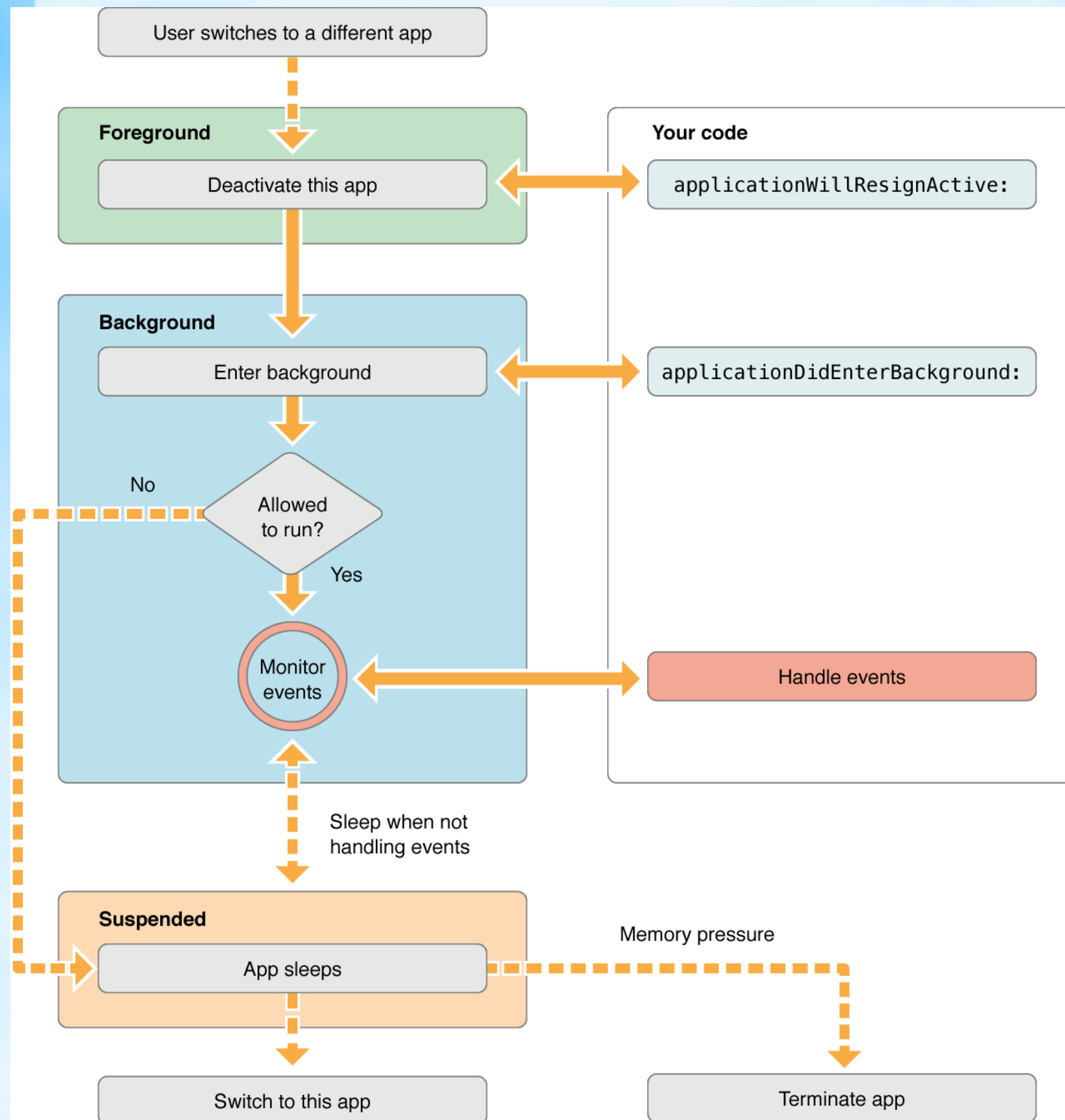
`applicationDidBecomeActive:`
executar mesmas tarefas de ativação
feitas na inicialização do app

Sistema de Notificação de mudança de estados

- *Um aplicativo no estado suspenso deve estar preparado para lidar com qualquer fila de notificação quando retorna para foreground ou background
- *Sistema junta todos os eventos e envia uma única notificação para que aplicativo não fique sobrecarregado
- *Lista de notificações:

Evento	Notifications
Acessório foi conectado ou desconectado	EAAccessoryDidConnectNotification EAAccessoryDidDisconnectNotification
A orientação do aparelho mudou	UIDeviceOrientationDidChangeNotification (view controllers atualizam sua interface automaticamente)
Existe uma mudança de tempo significativa	UIApplicationSignificantTimeChangeNotification
O estado ou nível de bateria mudou	UIDeviceBatteryLevelDidChangeNotification UIDeviceBatteryStateDidChangeNotification
O estado de proximidade foi alterado	UIDeviceProximityStateDidChangeNotification
O estado de arquivos protegidos foi alterado	UIApplicationProtectedDataWillBecomeUnavailable UIApplicationProtectedDataDidBecomeAvailable
Um display externo foi conectado ou desconectado	UIScreenDidConnectNotification UIScreenDidDisconnectNotification
O modo de display da tela foi alterado	UIScreenModeDidChangeNotification
Preferencias expostas pelo aplicativo nas configurações mudou	NSUserDefaultsDidChangeNotification
A língua atual ou configuração de localização foi alterada	NSCurrentLocaleDidChangeNotification
O status da conta de iCloud do usuário foi alterado	NSUbiquityIdentityDidChangeNotification

Transição Foreground - Background



* `applicationDidEnterBackground` deve executar as seguintes tarefas:

- * Preparar app para captura de tela (esconder informações sensíveis), captura é feita após o retorno do método
- * Salvar informações relevantes do estado do app
- * Liberar memória
- * A função tem aproximadamente 5 segundos para executar qualquer tarefa e retornar. Caso precisar de mais tempo deve-se chamar a função `beginBackgroundTaskWithExpirationHandler` para a solicitação de um tempo a mais de execução em background

Reduzindo o consumo de memória

- * Apps que consomem maior memória são os primeiros a serem terminados pelo sistema
- * Deve-se remover referências fortes (strong) dos objetos que não são mais necessários, alguns exemplos são:
 - * Objetos de imagem (UIImage)
 - * Mídias ou arquivos grandes que podem ser carregados novamente pelo disco
 - * Qualquer objeto que o aplicativo não precise que pode ser recriado facilmente.

Referências

*Apple. *App Programming Guide for iOS*. Disponível em: https://developer.apple.com/library/archive/documentation/iPhone/Conceptual/iPhoneOSProgrammingGuide/Introduction/Introduction.html#/apple_ref/doc/uid/TP40007072