

# Vòng đồi ứng dụng

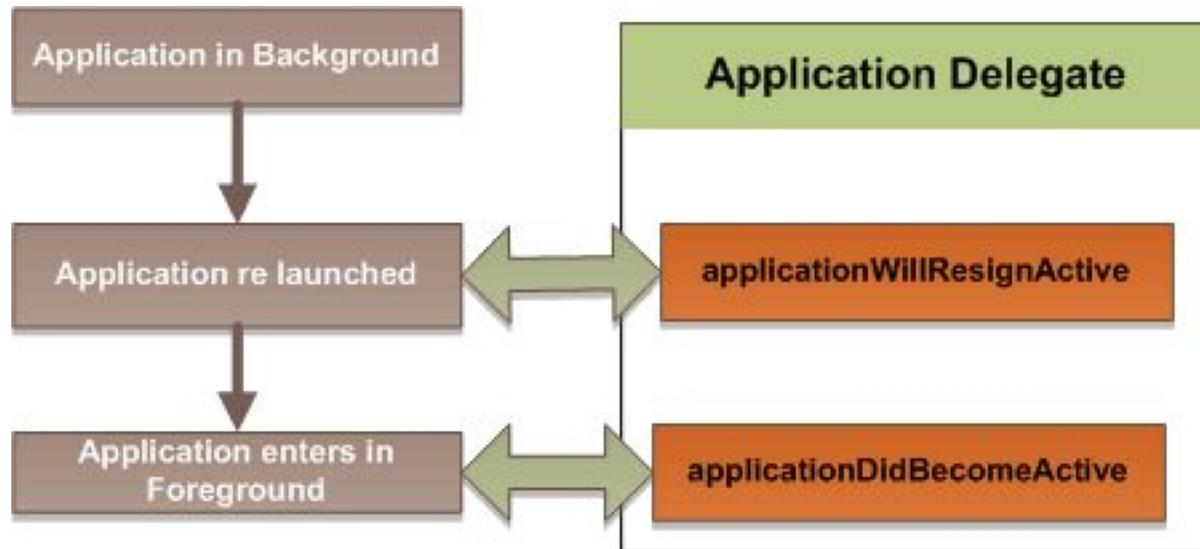


# Giới thiệu

---

- Vòng đời của 1 ứng dụng khá là quan trọng đối với các bạn developer - những người luôn muốn mang tới các trải nghiệm, phong phú, đa dạng và tốt nhất cho người dùng.

*Khi user mở điện thoại của mình lên thì không có ứng dụng nào chạy ngoài những thứ nằm trong **OS**. **App** của bạn cũng sẽ không chạy. Sau khi user nhấn mở, **Springboard** sẽ kích hoạt app của bạn. App cùng với các thư viện của nó sẽ được thực thi và được tải vào bộ nhớ, trong khi đó thì Springboard sẽ nhận nhiệm vụ hiển thị màn hình **launch screen** của app. Sau cùng thì app của bạn bắt đầu được chạy và **application delegate** sẽ nhận được các **notification***



# Trạng thái của app

---

- Mọi ứng dụng iOS chạy trên các thiết bị đều có các trạng thái chuyển đổi như: **Not running**, **In active**, **Active**, **Background**, **Suspended**. Tại bất kì thời điểm nào, app của bạn đều rơi vào các trạng thái trên:

Trạng thái	Điễn tả
<b>Not running</b>	
<b>Active (Foreground)</b>	<i>Trạng thái khi chương trình đang hoạt động bình thường</i>
<b>Inactive (Foreground)</b>	<i>Tại trạng thái Active, nếu chương trình nhận được các sự kiện từ hệ thống như có tin nhắn mới, có điện thoại, thì ứng dụng sẽ chuyển sang trạng thái Inactive</i>
<b>Background</b>	<i>Khi người dùng ấn phím Home, chương trình sẽ chuyển sang trạng thái background, ở trạng thái này ứng dụng hoàn toàn không hiển thị với người dùng, tuy nhiên các dòng lệnh vẫn được thực thi</i>
<b>Suspended</b>	<i>Ứng dụng nếu ở trạng thái background sau 5 - 10 giây thì sẽ bị "đóng băng" tạm thời, và ứng dụng chỉ thoát khỏi đóng băng khi người dùng chạy lại chương trình</i>

- Chúng ta có thể bắt được sự thay đổi trạng thái của app qua các hàm trong AppDelegate sau đây:

```
func application(_ application: UIApplication, didFinishLaunchingWithOptions launchOptions: [UIApplication.LaunchOptionsKey: Any]?) -> Bool {
    // Override point for customization after application launch.
    return true
}

func applicationWillResignActive(_ application: UIApplication) {
    // Sent when the application is about to move from active to inactive state. This can occur for certain types of temporary interruptions (such as an incoming phone call or SMS message) or when the user quits the application and it begins the transition to the background state.
    // Use this method to pause ongoing tasks, disable timers, and invalidate graphics rendering callbacks. Games should use this method to pause the game.
}

func applicationDidEnterBackground(_ application: UIApplication) {
    // Use this method to release shared resources, save user data, invalidate timers, and store enough application state information to restore your application to its current state in case it is terminated later.
    // If your application supports background execution, this method is called instead of applicationWillTerminate: when the user quits.
}

func applicationWillEnterForeground(_ application: UIApplication) {
    // Called as part of the transition from the background to the active state; here you can undo many of the changes made on entering the background.
}

func applicationDidBecomeActive(_ application: UIApplication) {
    // Restart any tasks that were paused (or not yet started) while the application was inactive. If the application was previously in the background, optionally refresh the user interface.
}

func applicationWillTerminate(_ application: UIApplication) {
    // Called when the application is about to terminate. Save data if appropriate. See also applicationDidEnterBackground:.
}
```

# Các hàm trong vòng đời ứng dụng

---

- **application:willFinishLaunchingWithOptions** --- Method này sẽ được gọi sau khi app đã khởi chạy thành công. Đây là method đầu tiên được gọi trong AppDelegate. Bạn có thể thực thi code trong hàm này khi việc khởi chạy đã hoàn thành.
- **application:didFinishLaunchingWithOptions** --- Method này được gọi trước khi window của app được hiển thị. Bạn có thể hoàn thiện giao diện của mình và cung cấp root viewcontroller cho window.

# Các hàm trong vòng đời ứng dụng

---

- **applicationDidBecomeActive** --- Method này được gọi để báo cho app của bạn biết khi nó chuyển trạng thái từ **In-Active** sang **Active** hoặc hệ thống và user khởi động app hoặc trong trường hợp user bỏ qua các gián đoạn làm app ngay lập tức chuyển sang In-Active(như là huỷ cuộc gọi hoặc tin nhắn). Nên dùng method này để chạy lại các tác vụ đang bị dừng(hoặc chưa chạy) khi app bắt đầu chạy lại.
- **applicationWillResignActive** --- Method này được gọi để báo cho app biết rằng nó sắp chuyển từ trạng thái Active sang In-Active . Nó xảy ra khi trường hợp bị gián đoạn (có cuộc gọi tới hoặc SMS hoặc báo thức) hay là khi user tắt app đi. Nên dùng method này để dừng các task đang chạy hoặc vô hiệu hóa timer trong app, hoặc nhiều thứ khác

# Các hàm trong vòng đời ứng dụng

---

- **applicationDidEnterBackground** --- Method này được gọi để báo cho app biết nó đang không chạy ở dưới foreground. Bạn có khoảng tầm 5 giây để thực thi các task . Trong trường hợp bạn muốn có nhiều thời gian hơn để xử lý, bạn có thể yêu cầu hệ thống cấp cho thời gian thực thi bằng cách gọi hàm **beginBackgroundTask(expirationHandler:)**. Nếu như method của bạn không được thực thi và trả về trước thời gian hết hạn thì app sẽ bị hệ thống chấm dứt và xoá khỏi bộ nhớ.

# Các hàm trong vòng đời ứng dụng

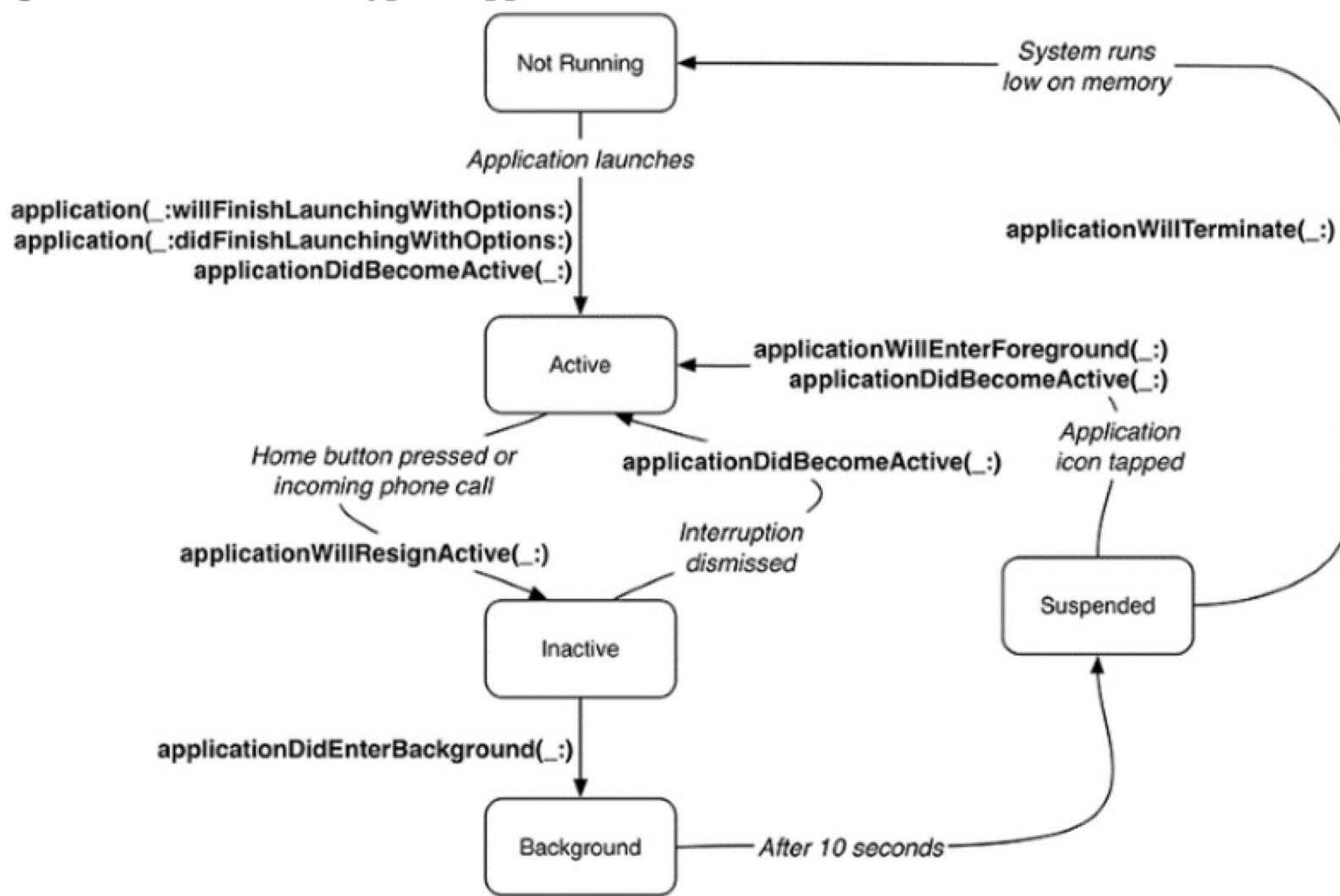
---

- **applicationWillEnterForeground** --- Method này được gọi như là 1 phần trong việc chuyển trạng thái từ **Background** sang **Active**. Nên dùng method này để hoàn thành các thay đổi đối với app trước khi nó xuống **Background**. **applicationDidBecomeActive** sẽ được gọi ngay khi method này đã hoàn thành việc chuyển trạng thái của app từ **In-Active** sang **Active**.

# Các hàm trong vòng đời ứng dụng

---

- **applicationWillTerminate** --- Method này được gọi khi app của bạn sắp bị hệ thống khai tử khỏi bộ nhớ. Bạn nên dùng method này để thực thi các tác vụ dọn dẹp. Bạn có tầm khoảng 5 giây để thực thi tác vụ. Nếu hàm của bạn không trả về trước thời gian hết hạn, hệ thống sẽ tự động khai tử app kèm cã task đang thực thi của bạn khỏi bộ nhớ. Method này cũng được gọi trong trường hợp app đang chạy ở dưới background( không bị suspended) nhưng hệ thống lại cần phải huỷ nó vì vài lí do gì đó. Bạn không nên đợi **applicationWillTerminate** được gọi rồi mới lưu lại data. Trong 1 vài trường hợp hi hữu, **applicationWillTerminate** sẽ không được gọi trước khi app bị khai tử (Vd như trong trường hợp máy của bạn reboot lại thì method này sẽ không được gọi).



# Vòng đời ViewController



# Giới thiệu

---

- **View Controller** là một đối tượng trong mô hình **MVC** (Model View Controller)
- Một trong những vai trò quan trọng của **View Controller** là quản lý các **View** hiển thị nội dung trong App
- **View Controller** có thể giao tiếp và kết hợp với các **View Controller** khác
- **View Controller** cung cấp nhiều hành vi (**behaviour**) phổ biến cho tất cả các ứng dụng iOS. Thông thường, những hành vi này được xây dựng vào các lớp cơ sở (**base class**).

# Các dạng View Controller

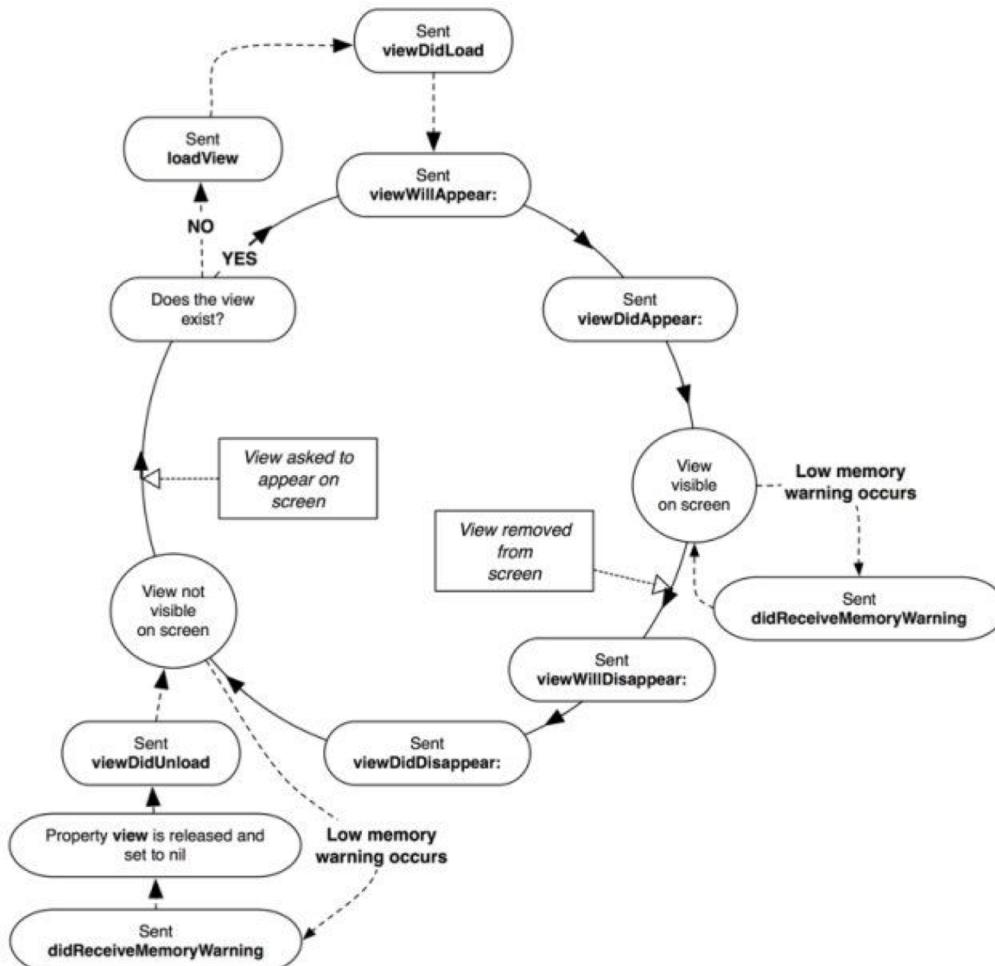
---

- **View Controller** có 2 dạng là **Content View Controller** và **Container View Controller**
- **Content View Controller** thể hiện nội dung trên màn hình bằng cách sử dụng một view hay một nhóm các view được tổ chức thành một hệ thống view  
Ví dụ: UITableViewController, CollectionViewController, ...
- **Container View Controller** là thành phần có khả năng chứa và quản lý nhiều View Controller bên trong.  
Ví dụ: TabBarController, Segment Controller

# Vòng đời View Controller

---

- Vòng đời của một View Controller là được tính từ lúc nó được nạp vào bộ nhớ (**RAM**) cho tới khi nó bị huỷ khỏi bộ nhớ.
- Để **huỷ View Controller**, chúng ta cần gọi hàm **deinit()**
- Sau đây là sơ đồ nói về hoạt động sống của một **View Controller** cùng với các hàm nó sẽ được gọi tự động tại một thời điểm xác định nào đó



# Trạng thái hoạt động của View Controller

---

## loadView

- Hàm này sẽ được gọi đầu tiên khi khởi tạo một *View Controller*.

## viewDidLoad

- Khi *View Controller* đã được nạp vào bộ nhớ, thì hàm **viewDidLoad** được gọi.
- Chỉ được gọi một lần duy nhất trong chu kỳ sống của view đó thôi
- Thường dùng để chuẩn bị data hoặc là khởi tạo các giá trị mặc định cho các object cũng như UI trên màn hình.

# Trạng thái hoạt động của View Controller

---

## **viewWillAppear**

- Hàm sẽ được gọi trước khi một view được thêm vào hệ thống view và trước animation hiển thị một view.

## **viewDidAppear**

- Hàm sẽ được gọi khi một view đã được thêm vào hệ thống view và đã hiển thị lên màn hình.

## **viewWillDisappear**

- Gọi khi một view đã được ẩn khỏi màn hình và animation khi ẩn view đó.

## **viewDidDisappear**

- Gọi khi một view đã được ẩn khỏi màn hình.

# **Lưu ý khi load dữ liệu tại các hàm:**

---

- Nếu dữ liệu là dạng tĩnh và **không thay đổi nhiều**, chúng ta nên load chúng ở phương thức **viewDidLoad**.
- Nếu dữ liệu là động và thay đổi thường xuyên, chúng ta nên load tại phương thức **viewDidAppear**.
- Lưu ý là tại 2 phương thức, dữ liệu cần phải load theo dạng **bất đồng bộ** ở một **thread** khác để tránh ảnh hưởng đến **UI**.

```
10
11 class ViewController: UIViewController {
12
13     override func viewDidLoad() {
14         super.viewDidLoad()
15         // Do any additional setup after loading the view.
16         print("viewDidLoad")
17     }
18
19     override func loadView() {
20         super.loadView()
21         print("load View")
22     }
23
24     override func viewWillAppear(_ animated: Bool) {
25         super.viewWillAppear(true)
26         print("view Will Appear")
27     }
28
29     override func viewDidAppear(_ animated: Bool) {
30         super.viewDidAppear(true)
31         print("view Did Appear")
32     }
33
34     override func viewWillDisappear(_ animated: Bool) {
35         super.viewWillDisappear(true)
36         print("view Will Disappear")
37     }
38
39     override func viewDidDisappear(_ animated: Bool) {
40         super.viewDidDisappear(true)
41         print("view Did Disappear")
42     }
43 }
```

MyProjectController

load View  
viewDidLoad  
view Will Appear  
view Did Appear

# Xcode 11, iOS 13

<https://techmaster.vn/posts/35579/app-delegate-va-scene-delegate-trong-xcode-11-va-ios-13>

