



Lập trình iOS

Bài 8. Split View Controller và Pop Over Controller

Ngành Mạng & Thiết bị di động





Nội dung

1. Split View Controller

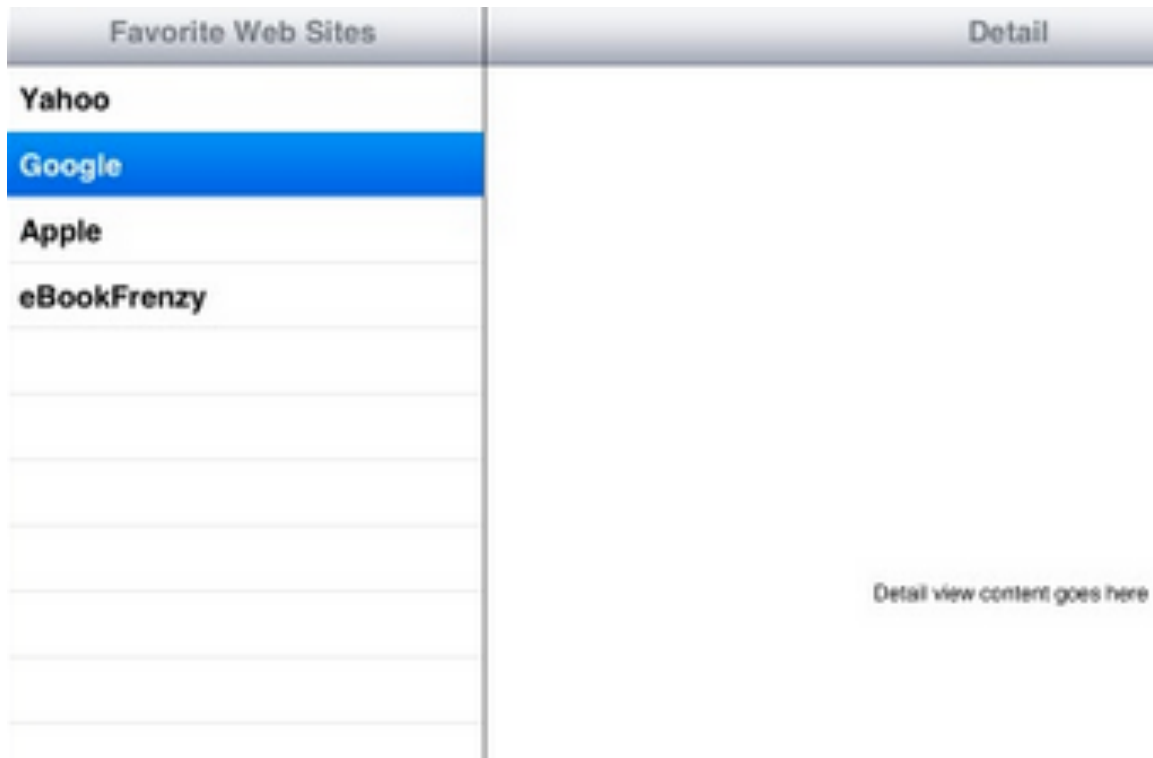
- Mục đích sử dụng trong ứng dụng
- Khảo sát lớp UISplitViewController
- UISplitViewControllerDelegate
- Xây dựng Split View Controller

2. Pop Over Controller



1.1 Mục đích sử dụng trong ứng dụng

- ❑ Split View được sử dụng nhằm hiển thị nội dung màn hình với một menu bên trái và nội dung bên phải.





1.1 Mục đích sử dụng trong ứng dụng

❑ Một số hình ảnh về Split View Controller.





1.1 Mục đích sử dụng trong ứng dụng

❑ Tổng quan.

- Mỗi panel được quản lý bởi một controller điều khiển riêng cho mình tùy thuộc vào mục đích sử dụng mà thiết kế cho phù hợp.
- Bên cạnh đó một số giao tác như quản lý quay màn hình được split view điều khiển thông qua delegate của nó.
- Một splitview controller phải là control gốc của các control còn lại. Nói cách khác splitview phải là root view của application's window trong ứng dụng của bạn.
- Mỗi panel có thể là navigation controller, tab bar controllers hoặc bất cứ loại view controller nào bạn cần.



1.1 Mục đích sử dụng trong ứng dụng

- Split view controllers không thể được sử dụng bằng cách presented.
- Cách đơn giản nhất để tạo một split view controller là khi tạo project mới. Chọn Split View-based Application template trong Xcode project mặc định sẽ bắt đầu bằng một split view controller. Việc còn lại là chỉ việc sử dụng các view controller khác cho panel left và right.



1.2 Khảo sát lớp UISplitViewController

❑ Thuộc tính quản lý View Controller con

- **viewControllers**: Một mảng các viewController được quản lý. Ở vị trí index 0 là view controller quản lý panel trái còn vị trí index 1 là view controller quản lý panel bên phải.
- **presentsWithGesture**: Kiểm tra xem split view có ẩn thông qua swipe gesture hay không. Mặc định là giá trị YES

❑ Thuộc tính delegate

- **delegate**: đối tượng nhận ủy thác từ Split View Controller.



1.3 UISplitViewControllerDelegate

❑ Trả lời các thay đổi chế độ hiển thị

- **splitViewController:willChangeToDisplayMode:** phương thức sẽ cho delegate biết được chế độ hiển thị khi có thay đổi.
- **targetDisplayModeForActionInSplitViewController:** phương thức sẽ yêu cầu delegate cung cấp chế độ hiển thị khi một hành động xảy ra trên split view controller.

❑ Cài đặt chuyển hướng

- - **splitViewControllerSupportedInterfaceOrientations:** Trả về các loại trạng thái mà Split View Controller hỗ trợ.
- - **splitViewControllerPreferredInterfaceOrientationForPresentation:** Trả về loại giá trị orientation mà Split View Controller đang hỗ trợ.



1.3 UISplitViewControllerDelegate

❑ Ẩn và hiện giao diện

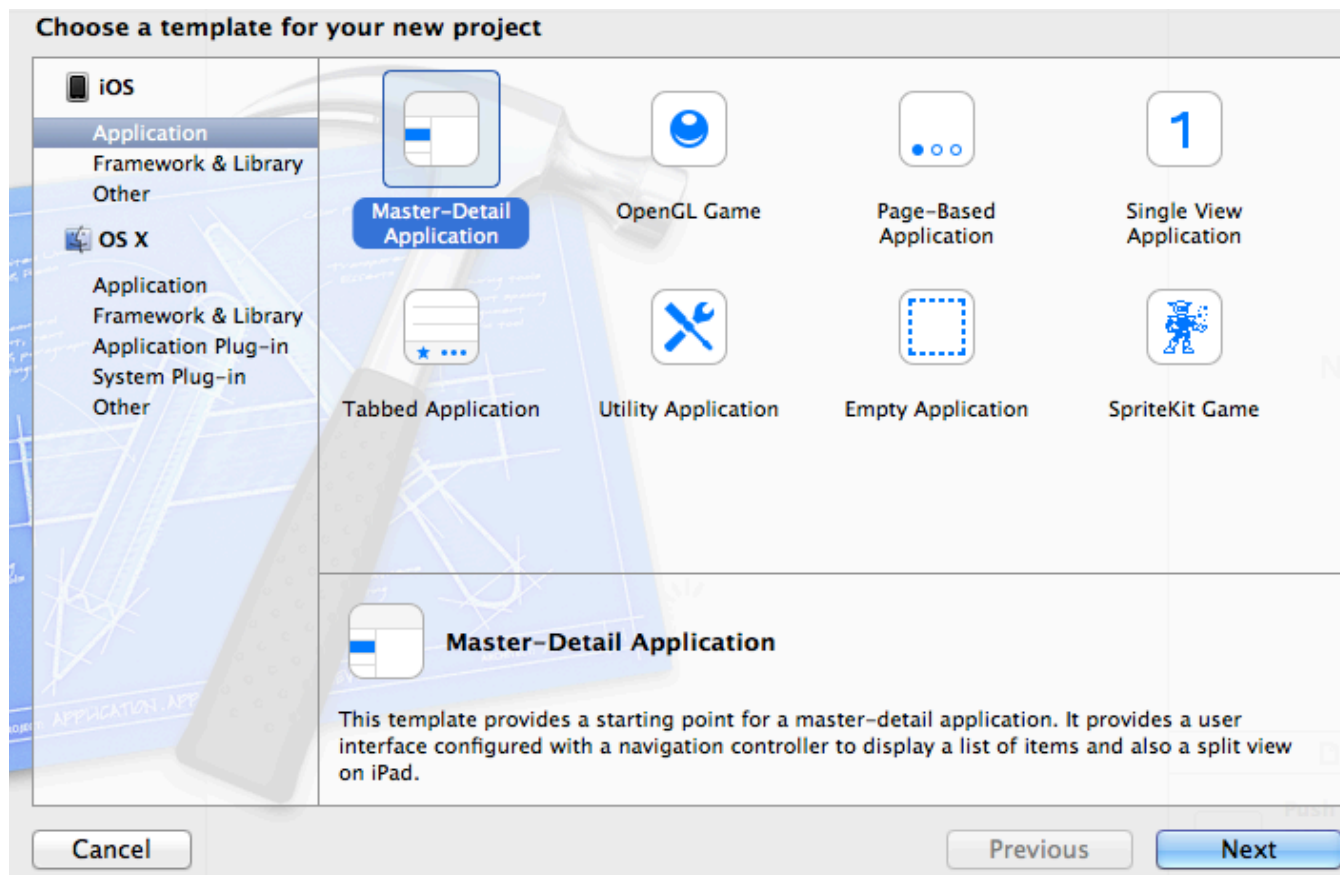
- - **primaryViewControllerForCollapsingSplitViewController**: Yêu cầu delegate cung cấp một view controller để hiển thị sau khi giao diện split view thu vào.
- - **splitViewController: collapseSecondaryViewController: ontoPrimaryViewController**: Yêu cầu delegate thiết lập view controller bên trái và để kết hợp view controller bên phải vào giao diện thu vào.
- - **primaryViewControllerForExpandingSplitViewController**: Yêu cầu delegate cung cấp một view controller danh sách bên trái khi split view hiện ra.
- - **splitViewController: separateSecondaryViewControllerFromPrimaryViewController**: Yêu cầu delegate cung cấp một view controller chi tiết bên phải.





1.4 Xây dựng Split View Controller

- ❑ Ta có thể dùng Split View Controller bằng cách chọn Master-Detail Application khi tạo project:

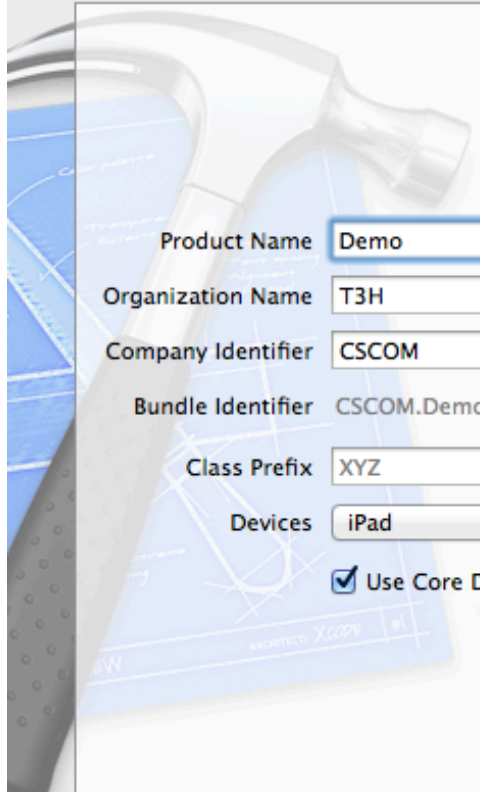




1.4 Xây dựng Split View Controller

❑ Tiếp theo chọn devide cho ipad :

Choose options for your new project:



Product Name

Organization Name

Company Identifier

Bundle Identifier

Class Prefix

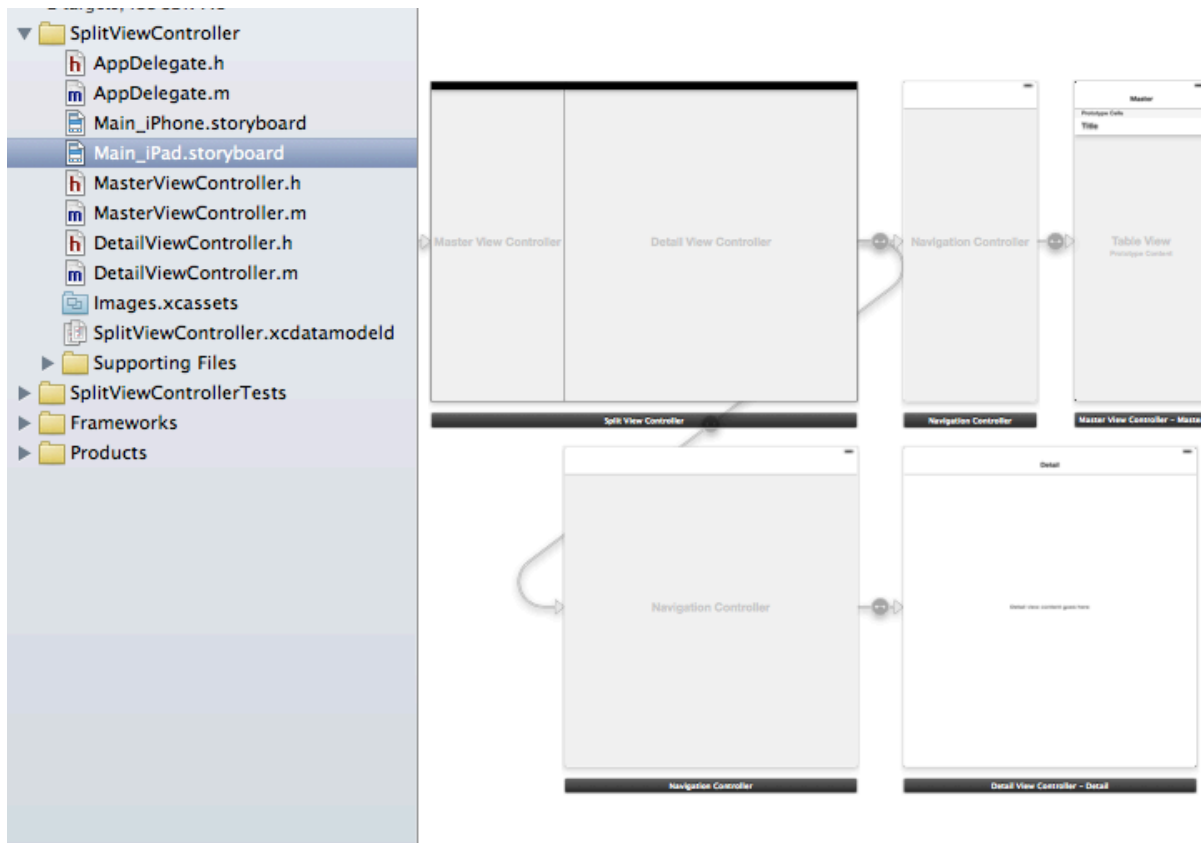
Devices

☒ Use Core Data



1.4 Xây dựng Split View Controller

- ❑ Cuối cùng project được tạo ra có giao diện gốc là một splitview :

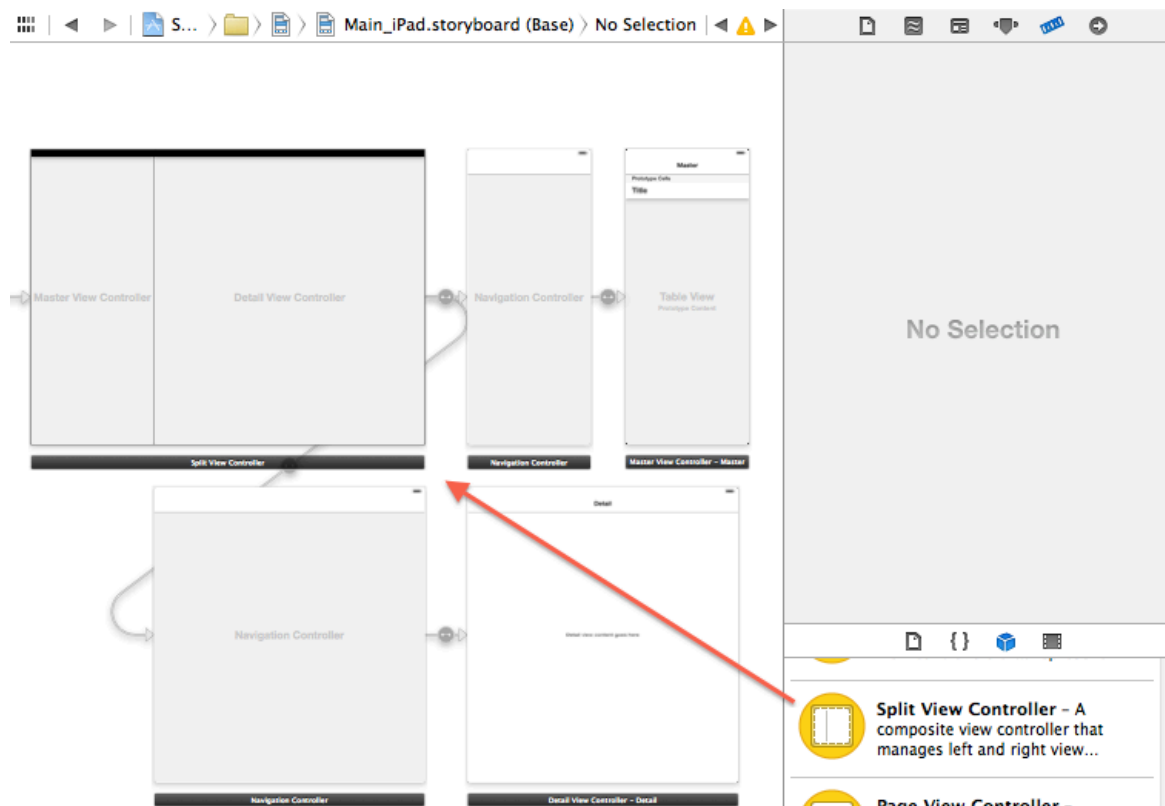




1.4 Xây dựng Split View Controller

❑ Xây dựng Split View Controller từ Interface Builder:

- Thực hiện kéo đối tượng UISplitViewController từ Object Library vào storyboard.





1.4 Xây dựng Split View Controller

❑ Xây dựng Split View Controller từ mã nguồn tại AppDelegate:

- Tạo ra đối tượng Navigation Controller và View Controller.

```
UINavigationController *menuNavigation =  
[[UINavigationController alloc] init];  
UITableViewController *menuTVC = [[UITableViewController  
alloc] init];  
[menuNavigation addChildViewController:menuTVC];  
  
UINavigationController *detailNavigation =  
[[UINavigationController alloc] init];  
UIViewController *detailVC = [[UIViewController alloc] init];  
[detailVC.view setBackgroundColor:[UIColor greenColor]];  
[detailNavigation addChildViewController:detailVC];
```



1.4 Xây dựng Split View Controller

- Tạo ra Split View Controller và thêm các navigation vừa tạo vào Split.

```
UISplitViewController *splitVC = [[UISplitViewController  
alloc]init];  
splitVC.viewControllers = [NSArray arrayWithObjects:  
menuNavigation, detailNavigation, nil];
```

- Thiết lập Split View Controller làm root view.

```
self.window.rootViewController =  
splitVC;
```



Nội dung

1. Split View Controller

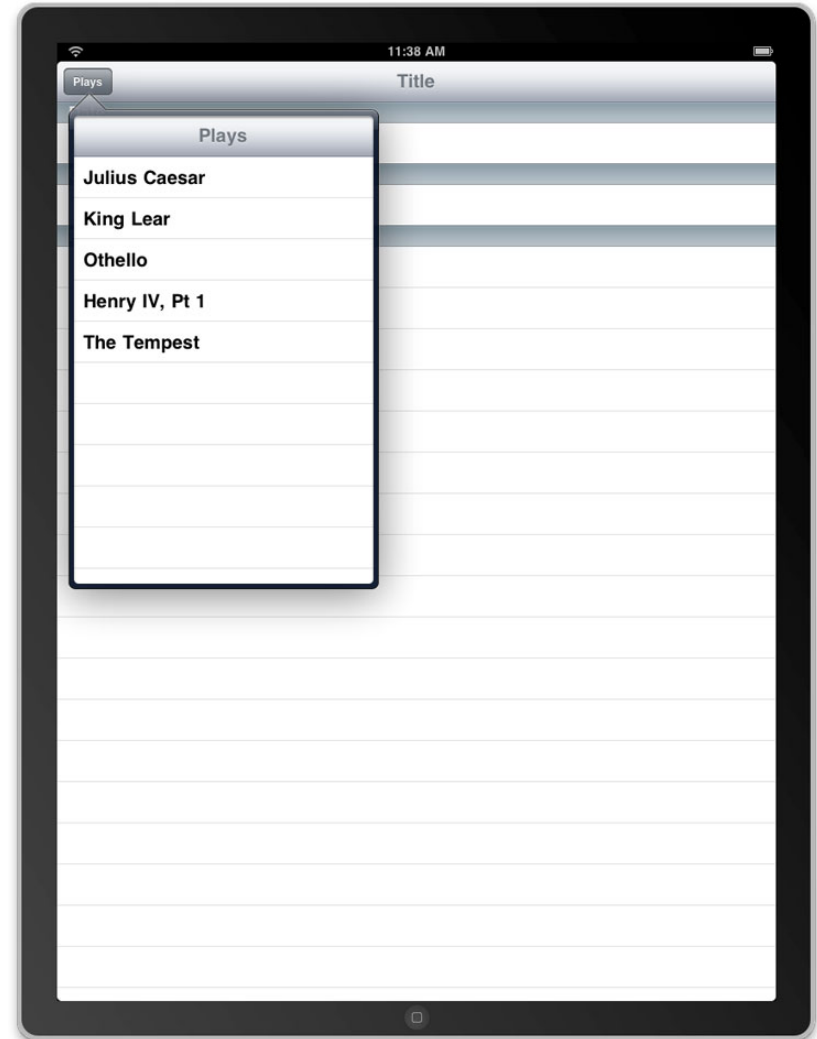
2. Pop Over Controller

- Giới thiệu
- Khảo sát lớp UIPopoverController
- UIPopoverControllerDelegate
- Xây dựng Popover Controller



2.1 Giới thiệu

- ❑ Popover Controller cung cấp một giao diện cửa sổ dùng để hiển thị nhanh trên một ViewController nhất định. Chỉ dùng trong ipad





2.2 Khảo sát lớp UIPopoverController

❑ Khởi tạo

- - **initWithContentViewController**: Phương thức khởi tạo với viewController gốc tương ứng.

❑ Hiện thị và đóng popover

- - **presentPopoverFromRect: inView:permittedArrowDirections: animated**: Hiện thị popover tại vị trí quy định.
- - **presentPopoverFromBarButtonItem: permittedArrowDirections: animated**: Hiện thị popover theo button bar quy định.
- - **dismissPopoverAnimated**: Loại bỏ một popover ra khỏi màn hình.



2.2 Khảo sát lớp UIPopoverController

❑ Cấu hình nội dung của popover

- **contentViewController**: Quy định ViewController quản lý nội dung của popover controller mà người dùng muốn sử dụng.
- - **setContentViewController:animated**: Thiết lập ViewController nội dung của popover controller.
- **popoverContentSize**: Quy định kích thước nội dung của popover.
- - **setPopoverContentSize:animated**: Thiết lập giá trị khung kích thước hiển thị nội dung của popover.
- **passthroughViews**: Khi thiết lập thông số này các View khác ngoài popover sẽ vẫn có thể tương tác.



2.2 Khảo sát lớp UIPopoverController

❑ Thuộc tính khác

- **popoverVisible**: Giá trị quy định popover đang được xuất hiện hay không. (Giá trị này chỉ đọc).
- **popoverArrowDirection**: Giá trị quy định hướng mũi tên của popover hiện tại. Giá trị mặc định là UIPopoverArrowDirectionUnknown.
- **backgroundColor**: Quy định màu nền của popover controller.
- **popoverBackgroundViewController**: Trả về lớp khi định quản lý nền của popover controller.



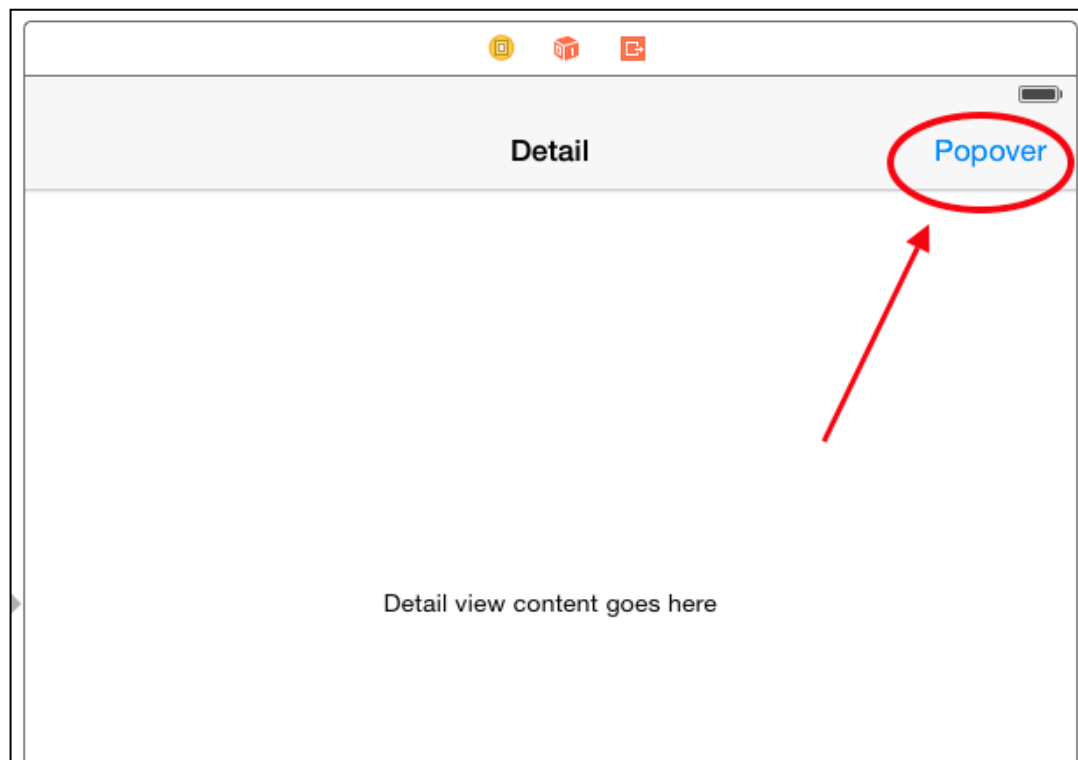
2.3 UIPopoverControllerDelegate

- - **popoverController:willRepositionPopoverToRect:inView:** Thông báo popover cần thay đổi vị trí trong view hiển thị.
- - **popoverControllerShouldDismissPopover:popoverController:** Yêu cầu loại bỏ popover ra khỏi khung nhìn.
- - **popoverControllerDidDismissPopover:** Thông báo popover đã bị loại bỏ ra khỏi màn hình.



3.4 Xây dựng Popover Controller

- ❑ Tiếp theo ví dụ của SplitViewController. Tạo một buttonbaritem:





3.4 Xây dựng Popover Controller

❑ Viết mã lệnh xử lý hiển thị Popover

```
- (IBAction)hienThiPopover:(id)sender {  
    MyViewController* content = [[MyViewController alloc] init];  
    UIPopoverController* aPopover = [[UIPopoverController alloc]  
initWithContentViewController:content];  
    [aPopover presentPopoverFromBarButtonItem:sender  
permittedArrowDirections:UIPopoverArrowDirectionAny  
animated:YES];  
}
```

