



# Lập trình iOS

## Bài 2. *ScrollView & Page Control*

Ngành Mạng & Thiết bị di động





# Nội dung

---

## 1. ScrollView

- Mục đích sử dụng trong ứng dụng
- Khảo sát lớp UIScrollView
- Xây dựng ScrollView
- Các tác vụ trong ScrollView

## 2. Page Control



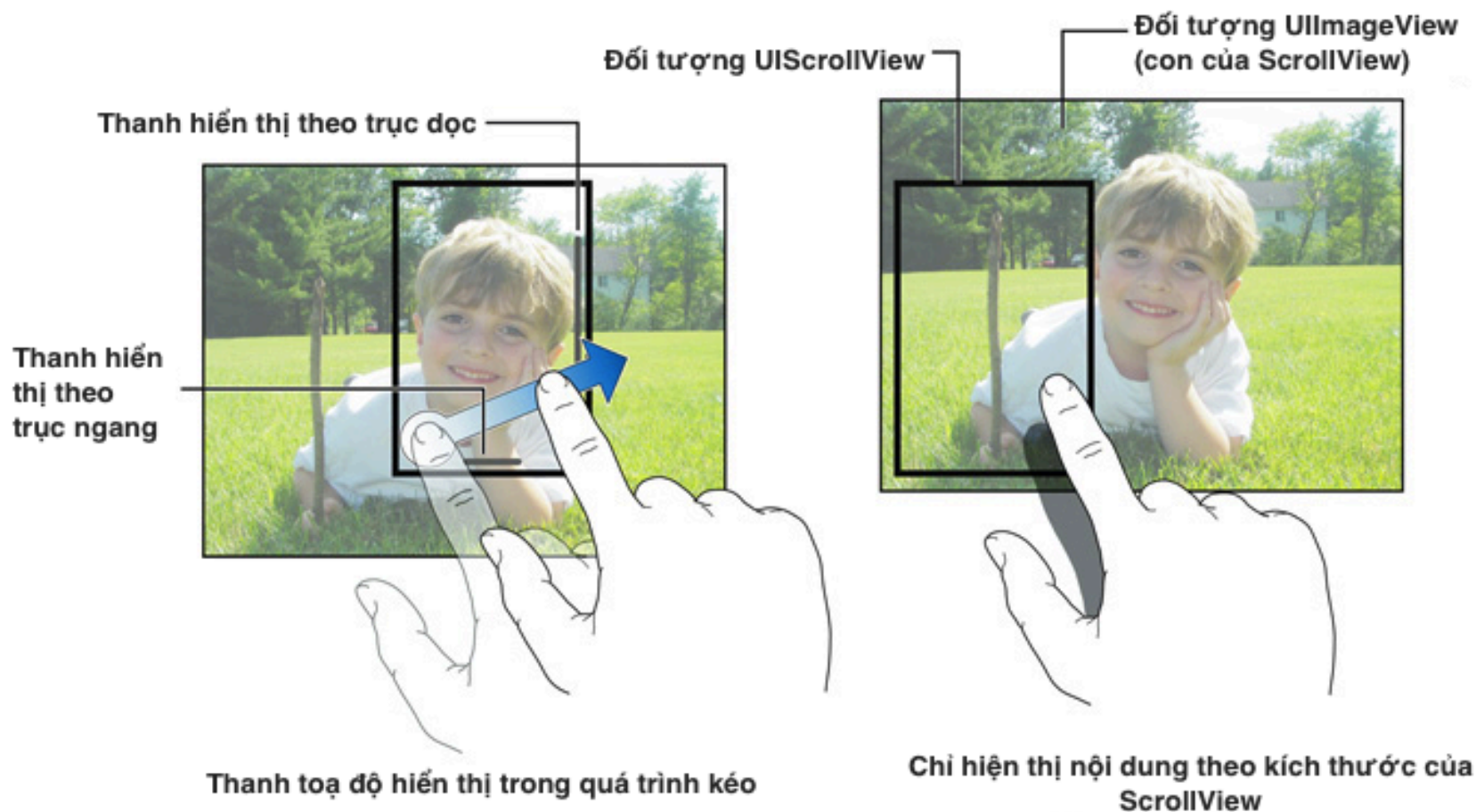
# 1.1 Mục đích sử dụng trong ứng dụng

---

- ❑ **ScrollView được sử dụng nhằm hiển thị các nội dung có kích thước hiển thị lớn hơn kích thước màn hình thiết bị. Bao gồm hai thao tác chính:**
  - Cho phép người dùng sử dụng thao tác kéo thả nội dung muốn hiển thị.
  - Thực hiện phóng to, thu nhỏ nội dung hiển thị.
- ❑ **ScrollView còn cho phép xây dựng hiển thị ở chế độ từng màn hình riêng lẻ (chế độ phân trang).**



# 1.1 Mục đích sử dụng trong ứng dụng





## 1.2 Khảo sát lớp UIScrollView

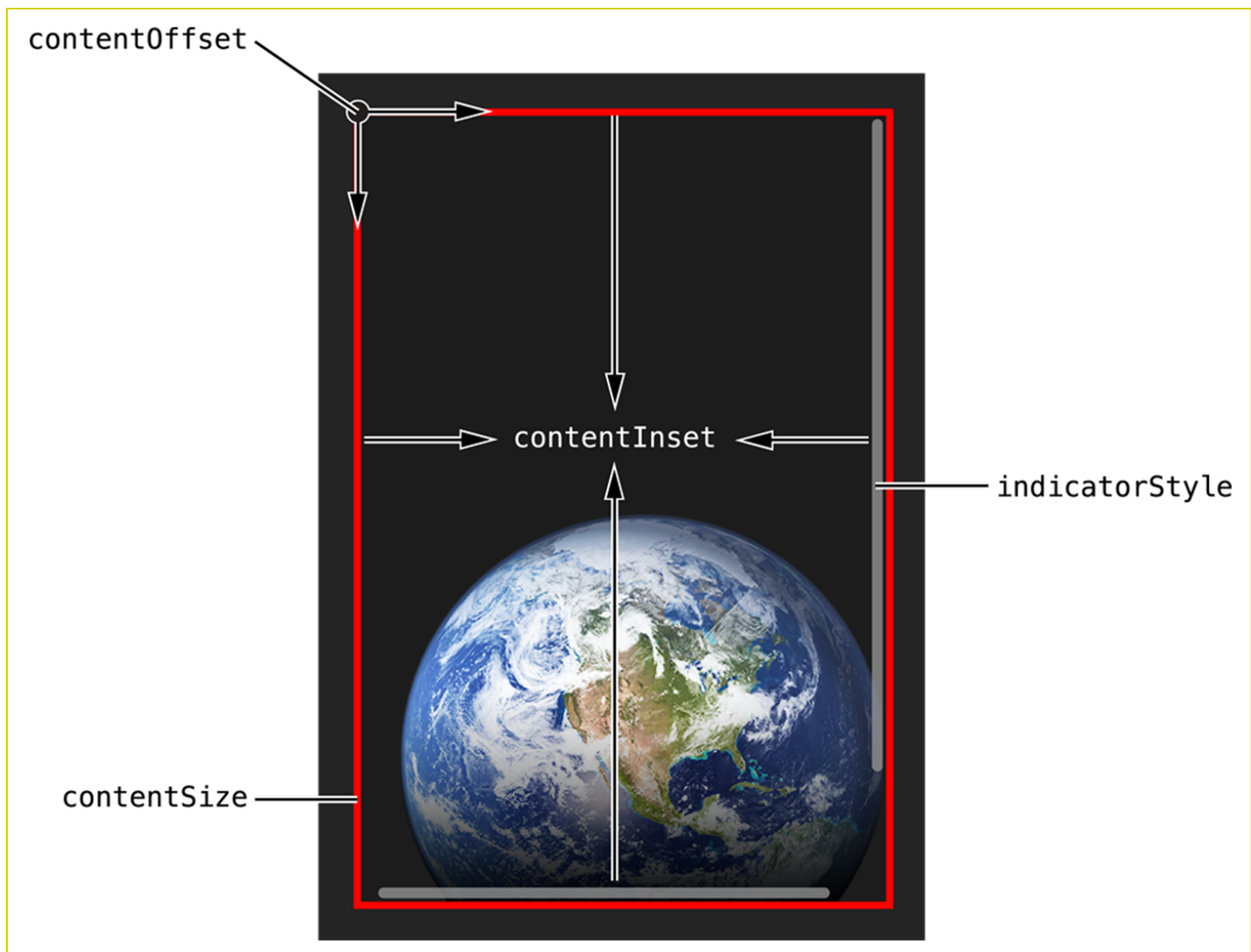
---

### ❑ Tùy chỉnh hiển thị nội dung

- - **setContentOffset:animated:** phương thức thiết lập kích thước (offset) cho nội dung hiển thị.
- **contentOffset:** thuộc tính điểm gốc của view chứa nội dung hiện tại trong scrollView
- **contentSize:** thuộc tính kích thước của contentView
- **contentInset:** thuộc tính khoảng cách bao từ nội dung đến các đường biên



## 1.2 Khảo sát lớp UIScrollView





## 1.2 Khảo sát lớp UIScrollView

---

### ❑ Tùy chỉnh các thao tác cuộn trong UIScrollView

- **scrollEnabled:** thuộc tính cho phép scrollView thực hiện thao tác cuộn.
- **directionalLockEnabled:** thuộc tính cho phép khóa thao tác cuộn ở một hướng nhất định.
- **bounces:** thuộc tính quản lý việc scroll view trượt tới đường biên và quay lại.



## 1.2 Khảo sát lớp UIScrollView

---

- ❑ **Tùy chỉnh các thông số thanh tọa độ trong UIScrollView**
  - **indicatorStyle**: thuộc tính chỉ định style của thanh cuộn
  - **showsHorizontalScrollIndicator**: thuộc tính cho phép hiển thị thanh cuộn theo chiều ngang
  - **showsVerticalScrollIndicator**: thuộc tính cho phép hiển thị thanh cuộn theo chiều dọc





## 1.2 Khảo sát lớp UIScrollView

---

### ❑ Tùy chỉnh các thao tác kéo thả trong UIScrollView

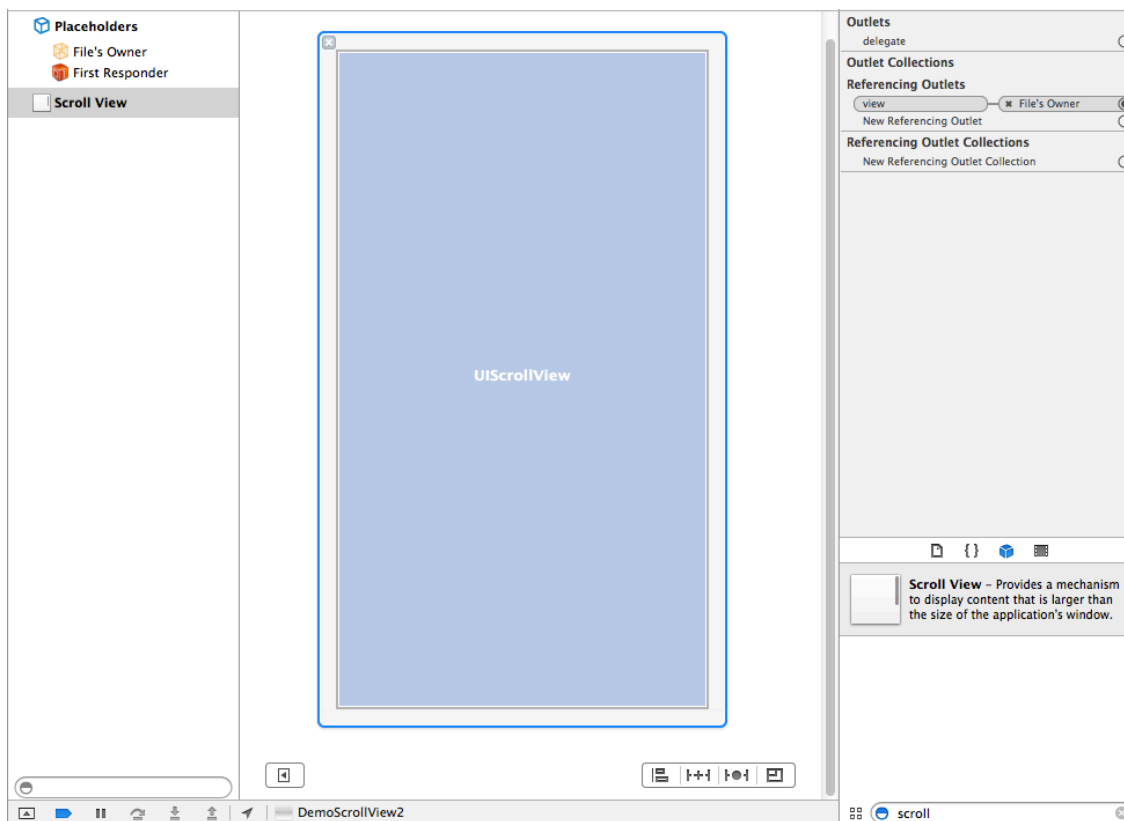
- **panGestureRecognizer**: thuộc tính quản lý bộ nhận dạng cử chỉ cho thao tác kéo đối tượng
- - **zoomToRect:animated**: phương thức hiển thị một khu vực nhất định trong scroll view
- **zoomScale**: thuộc tính quy định kích thước của scroll view khi phóng to hoặc thu nhỏ
- **maximumZoomScale**, **minimumZoomScale**: thuộc tính qui định kích thước maximum và minimum khi phóng to, thu nhỏ đối tượng trong scroll view

# 1.3 Xây dựng ScrollView



## ❑ Xây dựng ScrollView từ Interface Builder:

- Thực hiện kéo đối tượng UIScrollView từ Object Library vào xib, sau đó thực hiện tạo outlet cho ScrollView đến File's Owner.





## 1.3 Xây dựng ScrollView

### ❑ Xây dựng ScrollView từ Interface Builder:

- Tùy chỉnh các thông số Scrollview trong thẻ Attributes.

The image shows the 'Scroll View' section of the Attributes Inspector in Xcode. It contains the following settings:

- Style:** Default
- Scroll Indicators:**
  - ☒ Shows Horizontal Indicator
  - ☒ Shows Vertical Indicator
- Scrolling:**
  - ☒ Scrolling Enabled
  - ☐ Paging Enabled
  - ☐ Direction Lock Enabled
- Bounce:**
  - ☒ Bounces
  - ☐ Bounce Horizontally
  - ☐ Bounce Vertically
- Zoom:**
  - Min: 1
  - Max: 1
- Touch:**
  - ☒ Bounces Zoom
  - ☒ Delays Content Touches
  - ☒ Cancellable Content Touches
- Keyboard:** Do not dismiss



## 1.3 Xây dựng ScrollView

### ❑ Xây dựng ScrollView từ bằng mã nguồn:

- Trong hàm loadView của UIViewController thực hiện khởi tạo ScrollView.

```
// Truy xuất thông tin kích thước màn hình ứng dụng
CGRect fullScreenRect =[ [UIScreen mainScreen] applicationFrame];

// Khởi tạo đối tượng UIScrollView
// trong khu vực kích thước màn hình ứng dụng
scrollView=[[UIScrollView alloc] initWithFrame:fullScreenRect];

// Tùy chỉnh lại kích thước
scrollView.contentSize=CGSizeMake(720,1280);

// Thiết lập View của UIViewController là ScrollView
self.view=scrollView;
```



## 1.3 Xây dựng ScrollView

### ❑ Xây dựng ScrollView từ bằng mã nguồn:

- Tùy chỉnh các thông số hiển thị ScrollView thông qua các thuộc tính trong lớp UIScrollView.

```
// Thiết lập kích thước
scrollView.contentSize = CGSizeMake(720, 1280);

// Thiết lập màu nền
scrollView.backgroundColor = [UIColor grayColor];

// Tắt hiệu ứng nảy khi cuộn
scrollView.bounces = NO;

// Thực hiện canh lề
scrollView.contentInset = UIEdgeInsetsMake(64.0, 0.0, 44.0, 0.0);
```



## 1.4 Các tác vụ trong UIScrollView

---

### ❑ UIScrollViewDelegate

- Cho phép thực hiện các phương thức sự kiện được xây dựng để tương tác trực tiếp với lớp UIScrollView.
- Tùy chỉnh các tương tác trên UIScrollView bao gồm: thu phóng nội dung, tăng giảm tốc độ cuộn, các chuyển hoạt...



## 1.4 Các tác vụ trong ScrollView

---

### ❑ Thao tác cuộn và kéo:

- - **scrollViewDidScroll**: phương thức khi nhận sự kiện sau khi scroll nội dung bên trong scroll view
- - **scrollViewWillBeginDragging**: phương thức ghi nhận sự kiện khi bắt đầu scroll
- - **scrollViewWillBeginZooming: withView**: phương thức ghi nhận sự kiện bắt đầu Zoom
- - **viewForZoomingInScrollView**: phương thức gọi view cần thực hiện tác vụ zoom trong scroll view



## 1.4 Các tác vụ trong ScrollView

---

### ❑ Thao tác thu phóng nội dung:

- - **viewForZoomingInScrollView**: trả về đối tượng view dùng để thu phóng trong scroll view
- - **scrollViewWillBeginZooming:withView**: phương thức sẽ được gọi khi bắt đầu thao tác thu phóng đối tượng view bên trong scroll view
- - **scrollViewDidEndZooming:withView:atScale**: phương thức sẽ được gọi khi kết thúc thao tác thu phóng đối tượng view bên trong scroll view
- - **scrollViewDidZoom**: phương thức sẽ được gọi khi thông số thu phóng thay đổi





# Nội dung

---

## 1. ScrollView

## 2. Page Control

- Mục đích sử dụng trong ứng dụng
- Khảo sát lớp UIPageControl
- Xây dựng Page Control



## 2.1 Mục đích sử dụng trong ứng dụng

- ❑ PageControl bao gồm một hàng các dấu chấm biểu tượng, với mỗi dấu sẽ đại diện cho một trang hoặc một màn hình trong ứng dụng có nhiều màn hình.





## 2.1 Mục đích sử dụng trong ứng dụng

---

- ☐ PageControl không bao gồm các nội dung hiển thị trong trang, đồng thời không quản lý các nội dung bên trong trang.
- ☐ Hiển thị cho biết trang hiện tại.
- ☐ Cho phép thực hiện chuyển qua các trang khác nhau thông qua tương tác người dùng.
- ☐ Kết hợp với các điều khiển khác như ScrollView – hiển thị chế độ phân trang.



## 2.2 Khảo sát lớp **UIPageControl**

---

- ❑ **Lớp `UIPageControl` bao gồm các thuộc tính cho phép thiết lập các thuộc tính và hành động cho điều khiển Page Control.**
  
- ❑ **Quản lý điều khiển trang.**
  - **`currentPage`**: thuộc tính cho biết trang hiện tại, thể hiện qua chấm màu trắng trên giao diện người dùng
  - **`numberOfPages`**: thuộc tính cho biết số lượng trang của `PageControl`
  - **`hidesForSinglePage`**: thuộc tính cho biết `PageControl` sẽ ẩn đi trong trường hợp chỉ có một trang



## 2.2 Khảo sát lớp `UIPageControl`

---

### ❑ Cập nhật nội dung hiển thị trang

- **`pageIndicatorTintColor`**: thuộc tính màu cho các dấu chấm phân trang.
- **`currentPageIndicatorTintColor`**: thuộc tính màu cho dấu chấm hiển thị trang hiện tại.

### ❑ Điều chỉnh số lượng trang

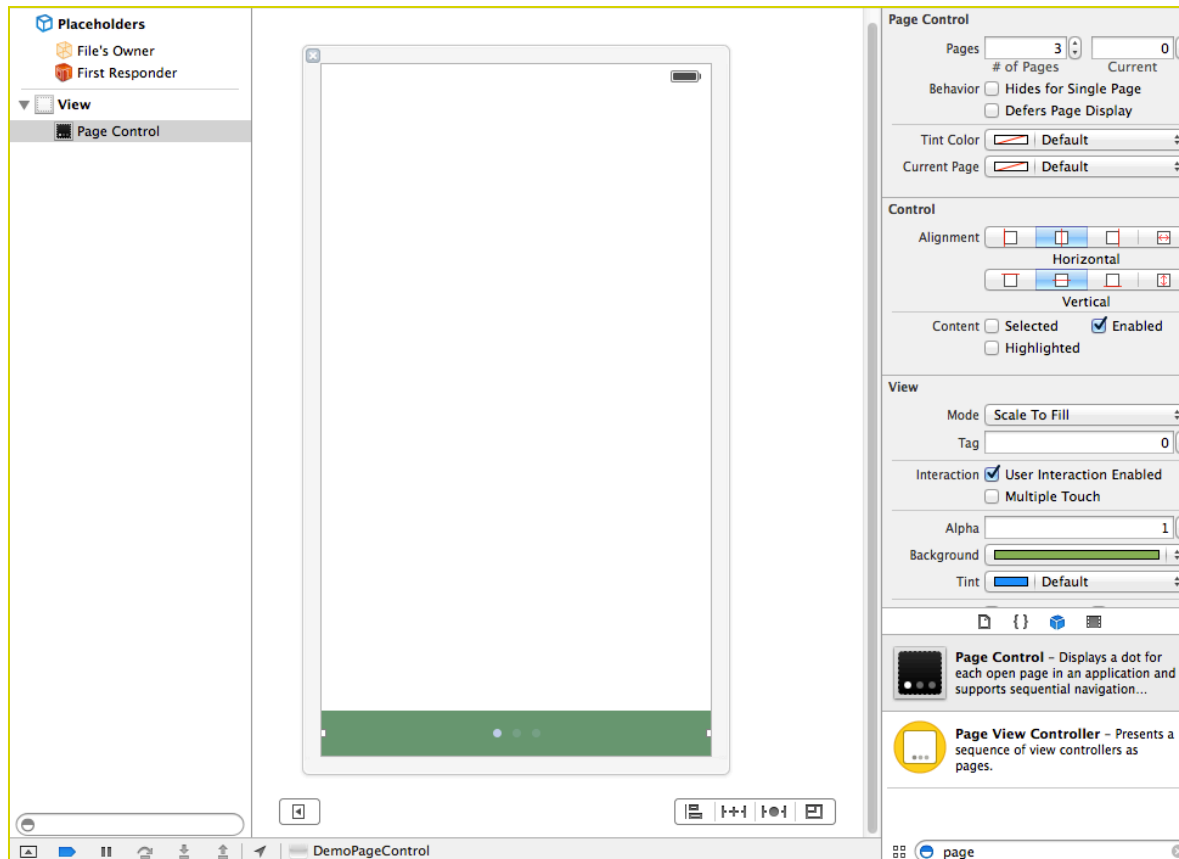
- - **`sizeForNumberOfPages`**: phương thức trả về kích thước tối thiểu để hiển thị cho số trang truyền vào

# 1.3 Xây dựng ScrollView



## ❑ Xây dựng Page Control từ Interface Builder:

- Thực hiện kéo đối tượng UIPageControl từ Object Library vào xib, sau đó thực hiện tạo outlet cho PageControl đến File's Owner.





## 1.3 Xây dựng ScrollView

### ❑ Xây dựng Page Control từ Interface Builder:

- Tùy chỉnh các thuộc tính trong thẻ Attributes

The screenshot shows the 'Page Control' attributes panel. It includes two numeric fields for 'Pages' (set to 3) and 'Current' (set to 0), each with up and down arrows. Below these are two unchecked checkboxes: 'Hides for Single Page' and 'Defers Page Display'. At the bottom, there are two color pickers, both labeled 'Default', for 'Tint Color' and 'Current Page'.



## 2.2 Xây dựng Page Control

- ❑ Có thể bắt lại sự kiện chuyển trang thông qua hành động `valueChange`.

- Ví dụ:

```
- (IBAction)changePage:(id)sender {
    NSInteger num = [self.pageControl currentPage];
    NSString* numString = [[NSString alloc] initWithFormat:
@"%d", ++num];
    [self.numberOfPage setText:numString];
    [self.pageControl updateCurrentPageDisplay];
}
```



