



Lập trình iOS

Bài 1. Khai thác tài nguyên Internet *WebView - Indicator*

Ngành Mạng & Thiết bị di động





Nội dung

1. Internet

- Khái niệm
- Lợi ích
- Các tài nguyên trên Internet

2. Networking và Internet trên iOS

3. WebView

4. Indicator



1.1. Khái niệm Internet

- ❑ Internet có nguồn gốc từ một dự án của Bộ Quốc phòng Mỹ tên là ARPANET được thực hiện vào những năm 70.
- ❑ Internet là mạng toàn cầu sử dụng giao thức TCP/IP. Internet không có chủ và không có mạng nào điều hành mạng nào.
- ❑ Người tham gia mạng có khả năng khai thác tài nguyên (resources) thông tin thậm chí các thiết bị trên các mạng thành viên khác và cũng có nghĩa vụ tạo ra các nguồn tài nguyên cho người khác sử dụng.



1.2. Lợi ích của Internet

- ☐ Email
- ☐ World Wide Web
- ☐ Search Engine
- ☐ Social Network



1.3. Các tài nguyên Internet

- ❑ Khái niệm tài nguyên (resource) chỉ những phương tiện, nguồn thông tin tiềm tàng, sẵn sàng để khai thác. Tài nguyên trên Internet có thể là thiết bị nhưng chủ yếu là thông tin.



1.3. Các tài nguyên Internet

- ❑ **Các dạng thông tin có thể lấy trên internet bao gồm**
 - Các văn bản (dạng TeX, Poscript. Các sách điện tử dạng PDF)
 - Các ảnh (ví dụ như các thư viện hội hoạ)
 - Các tài liệu lưu trữ dưới dạng âm thanh (các bản nhạc, lời nói...)
 - Phim video số
 - Các phần mềm máy tính.
 - Web, Hypertext
 - Máy tính từ xa → tài nguyên có thể là thiết bị



Nội dung

1. Internet

2. Networking và Internet trên iOS

- Game Center
- BSD Socket
- Webkit Framework
- Kiểm tra kết nối Internet trên iOS app

3. WebView

4. Indicator



2.1. Game Center

- ❑ Các trò chơi trên iOS hay OS X đều có thể kết nối vào Game Center do Apple cung cấp.
- ❑ Người chơi có thể chia sẻ điểm số tốt nhất của họ, so sánh thành tích với các đối thủ khác, mời bạn bè tham gia trò chơi multilayer cũng như hệ thống tự động tìm người chơi.



2.1. Game Center



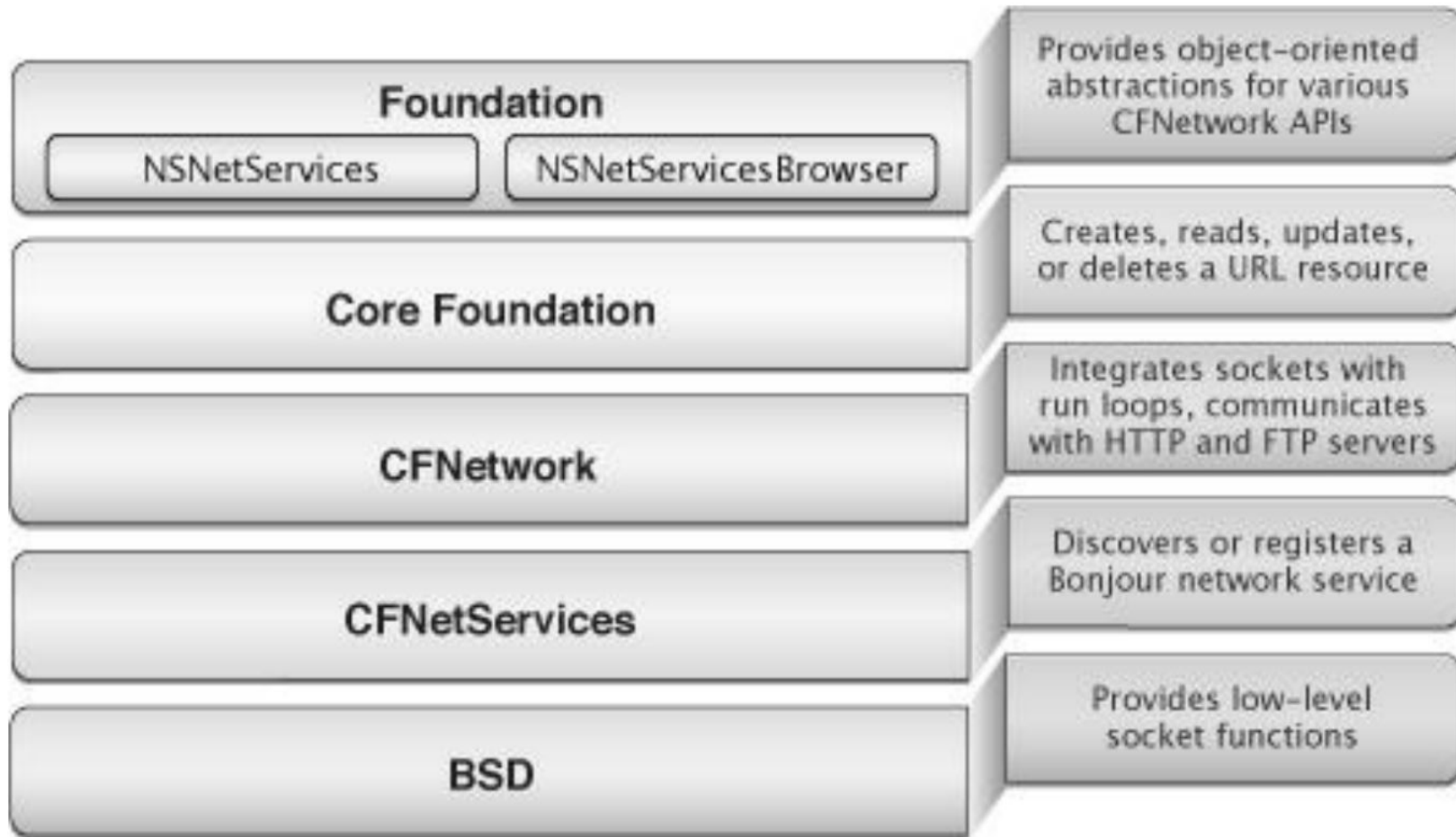


2.2. BSD Socket

- ❑ BSD Socket là một trong những nền tảng lập trình mạng cơ bản nhất của iOS.
- ❑ Hầu như các network framework khác đều phát triển dựa trên nền tảng của nó.
- ❑ BSD Socket được hỗ trợ trong iOS thỏa mãn các tính chất tiêu chuẩn của UNIX network programming.



2.2. BSD Socket





2.3. Webkit Framework

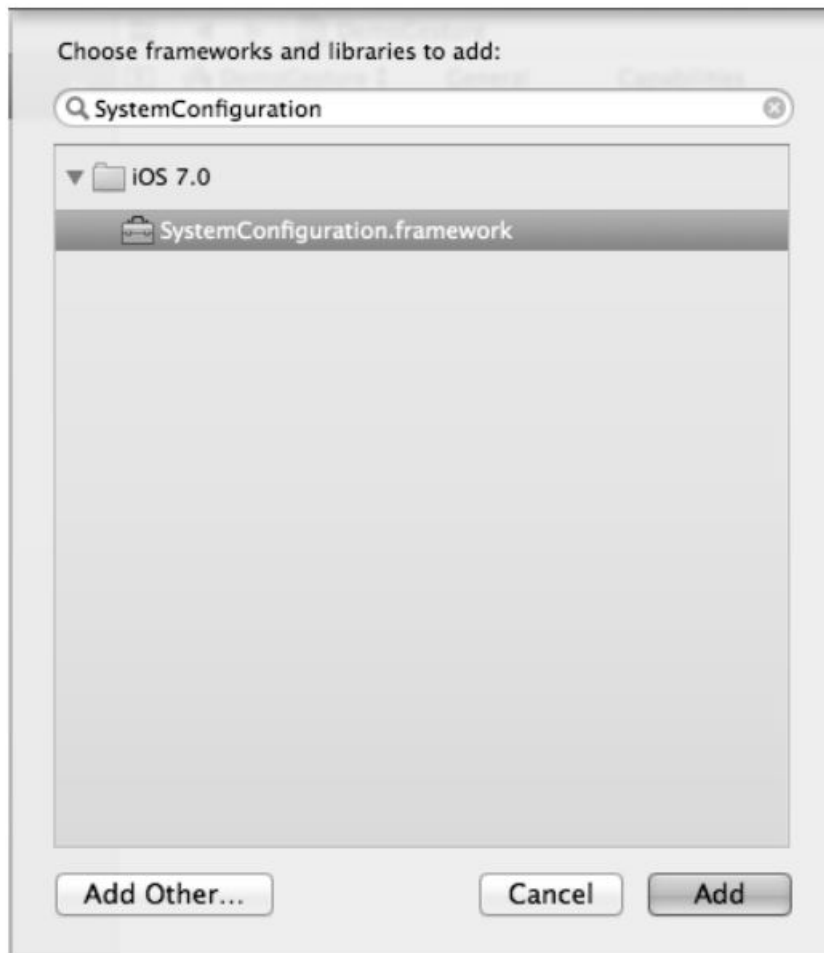
❑ Webkit được xem là một framework cấp cao cung cấp cho chúng ta đầy đủ các tính năng, khả năng kiểm soát các nội dung HTML, lưu trữ cookies, cache, thông tin quan trọng và xác thực người dùng.





2.4. Kiểm tra kết nối Internet trên iOS app

❑ Import thư viện SystemConfiguration





2.4. Kiểm tra kết nối Internet trên iOS app

- ❑ Việc kiểm tra kết nối internet thông qua Wifi, 4G, 3G, EDGE... được apple cung cấp một thư viện nhỏ để giải quyết vấn đề trên, đó là: “Reachability”. Ta vào đường dẫn bên dưới để tải thư viện Reachability:

https://developer.apple.com/library/ios/samplecode/Reachability/Listings/Reachability_APLViewController_m.html



2.4. Kiểm tra kết nối Internet trên iOS app

- ❑ Import Reachability.h và thư viện SystemConfiguration vào tập tin ViewController.h

```
#import <UIKit/UIKit.h>
#import "Reachability.h"
#import <SystemConfiguration/SystemConfiguration.h>
```



2.4. Kiểm tra kết nối Internet trên iOS app

- ❑ Viết đoạn mã lệnh kiểm tra kết nối internet trong viewDidLoad

```
- (void)viewDidLoad {
    [super viewDidLoad];
    Reachability *reachability= [Reachability
    reachabilityForInternetConnection];
    NetworkStatus  networkStatus =[ reachability
    currentReachabilityStatus];
    if (networkStatus == ReachableViaWWAN) {
        // Trường hợp kết nối internet qua WAN
        NSLog(@"Trường hợp kết nối internet qua WAN!");
    } else if (networkStatus == ReachableViaWiFi){
        // Trường hợp kết nối internet qua WiFi
        NSLog(@"Trường hợp kết nối internet qua WiFi!");
    } else if (networkStatus == NotReachable){
        // Không có kết nối internet
        NSLog(@"Không có kết nối internet!");
    }
}
```




Nội dung

1. Internet

2. Networking và Internet trên iOS

3. WebView

- Giới thiệu về WebView
- Khảo sát lớp UIWebView
- UIWebViewDelegate
- Ví dụ

4. Indicator

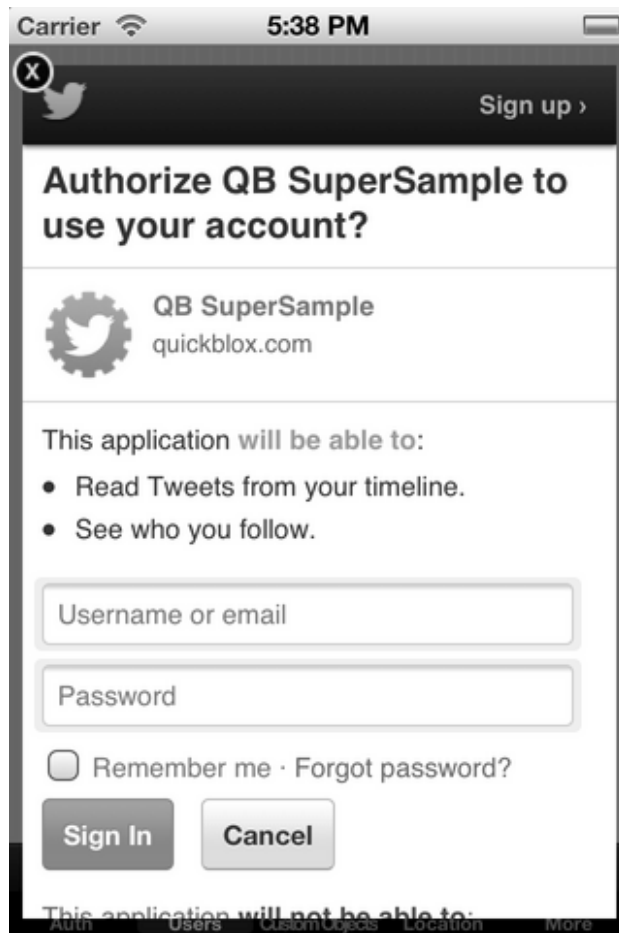


3.1. Giới thiệu về WebView

- ❑ Sử dụng `UIWebView` để nhúng một nội dung trang web vào ứng dụng.
- ❑ Có thể quản lý lịch sử, nội dung và một số tính chất của trang web.
- ❑ Lưu ý : Không nên sử dụng `UIWebView` hoặc `UITableView` ở trong `UIScrollView` sẽ dễ dẫn tới việc đung độ các scroll.



3.1. Giới thiệu về WebView



Webview đăng nhập của Twitter



3.1. Giới thiệu về WebView

❑ UIWebView có thể hỗ trợ đọc một số file văn bản có định dạng như sau :

- Excel (.xls)
- Keynote (.key.zip)
- Numbers (.numbers.zip)
- Pages (.pages.zip)
- PDF (.pdf)
- Powerpoint (.ppt)
- Word (.doc)
- Rich Text Format (.rtf)
- Rich Text Format Directory (.rtfd.zip)
- Keynote '09 (.key)
- Numbers '09 (.numbers)
- Pages '09 (.pages)



3.2. Khảo sát lớp UIWebView

❑ Quản lý nội dung

- **-loadData:MIMETYPE:textEncodingName:baseURL:** Hiển thị nội dung với định dạng MIME, mã hóa encoding.
- **-loadHTMLString :baseURL:** Hiển thị nội dung trang chính với chuỗi và theo URL cơ sở
- **-loadRequest:** Kết nối hiển thị nội dung tương ứng với URLRequest theo dạng không đồng bộ
- **request** :Lấy ra URL đang được tải nội dung (chỉ cho phép đọc)
- **loading** : Thuộc tính kiểm tra webview có đang tải nội dung không. Nếu YES là đang tải nội dung, NO là ngược lại
- **- stopLoading** : Yêu cầu dừng tải nội dung của webview
- **- reload** : Tải lại trang hiện tại



3.2. Khảo sát lớp UIWebView

❑ Quản lý chuyển tiếp và quay lại

- **canGoBack**: Quy định webview có thể quay trở lại trang trước hay không (chỉ cho phép đọc).
- **canGoForward**: Quy định webview có thể di chuyển lại trang phía trước hay không (chỉ cho phép đọc)
- **-goBack**: Tải lại vị trí trang trước đó
- **-goForward**: Tải lại vị trí trang trước khi thực hiện goBack



3.2. Khảo sát lớp UIWebView

❑ Cài đặt nội dung hiển thị

- **scalesPageToFit**: Quy định nội dung hiển thị nó đúng bằng diện tích của webview không
- **scrollView**: Quản lý scrollview của webview (chỉ cho phép đọc).
- **keyboardDisplayRequiresUser**: Thiết lập có hiện thị bàn phím đối với một số nội dung trong webview hay không.
- **detectsPhoneNumbers**: Quy định có phát hiện và chuyển số điện thoại thành dạng liên kết number

❑ Cài đặt JavaScript

- **-stringByEvaluatingJavaScriptFromString**: Trả về kết quả chạy của một đoạn lệnh javascript



3.2. Khảo sát lớp UIWebView

❑ Cài đặt nội dung hiển thị

- **scalesPageToFit**: Quy định nội dung hiển thị nó đúng bằng diện tích của webview không
- **scrollView**: Quản lý scrollview của webview (chỉ cho phép đọc).
- **keyboardDisplayRequiresUser**: Thiết lập có hiện thị bàn phím đối với một số nội dung trong webview hay không.
- **detectsPhoneNumbers**: Quy định có phát hiện và chuyển số điện thoại thành dạng liên kết number

❑ Cài đặt JavaScript

- **-stringByEvaluatingJavaScriptFromString**: Trả về kết quả chạy của một đoạn lệnh javascript



3.3. UIWebViewDelegate

- **-webView:shouldStartLoadWithRequest:navigationType:** Thông báo webview bắt đầu tải nội dung theo hành động.
- **-webViewDidStartLoad:** webView đã bắt đầu tải nội dung.
- **-webViewDidFinishLoad:** webView đã tải xong nội dung.
- **-webViewdidFailLoadWith Error:** Có lỗi trong quá trình tải trang



3.4. Ví dụ

- ☐ Load nội dung từ link online
- ☐ Load nội dung từ chuỗi HTML
- ☐ Load javascript cho webview



Nội dung

1. Internet

2. Networking và Internet trên iOS

3. WebView

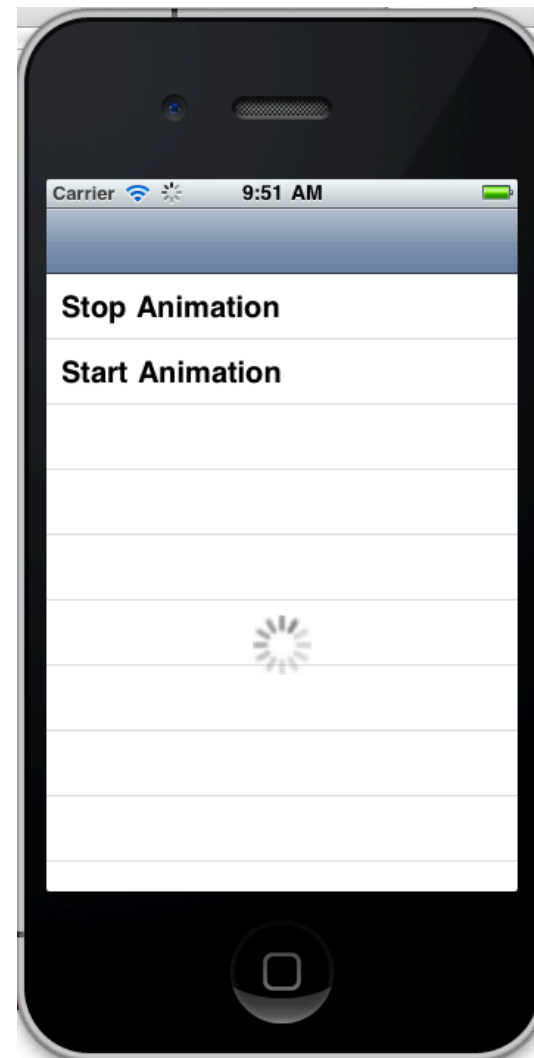
4. Indicator

- Giới thiệu về UIActivityIndicatorView
- Khảo sát lớp UIActivityIndicatorView
- Ví dụ



4.1. Giới thiệu UIActivityIndicatorView

- ❑ **UIActivityIndicatorView** là một control giúp hiển thị và báo rằng công việc đang được thực hiện.





4.2. Một số thuộc tính

❑ Khởi tạo

- **-initWithActivityIndicatorStyle:** Khởi tạo và trả về đối tượng `ActivityIndicator` theo style truyền vào

❑ Quản lý indicator

- **-startAnimating:** Bắt đầu animation cho indicator.
- **-stopAnimating:** Dừng animation cho indicator.
- **-isAnimating:** Kiểm tra animation có đang hoạt động không.
- **hidesWhenStopped:** Nếu là YES indicator sẽ tự động ẩn hiển thị khi `stopAnimating`.



4.2. Một số thuộc tính

❑ Cấu hình indicator

- **activityIndicatorViewStyle**: Thiết lập style cho indicator theo một số style có sẵn. Mặc định sẽ là `UIActivityIndicatorViewStyleWhite`.
- **color**: Thiết lập màu của indicator.



4.3. Ví dụ

- ❑ Sử dụng indicator để hiển thị trong quá trình load một URL trên Web View

