

# iPhone App Development

CM420-09-2016-C  
Lesson 8

Lecturer

**Bryant Tang**

bryant.tang14mo@gmail.com

CPTTMLAB\_B  
pw: cpttm1234

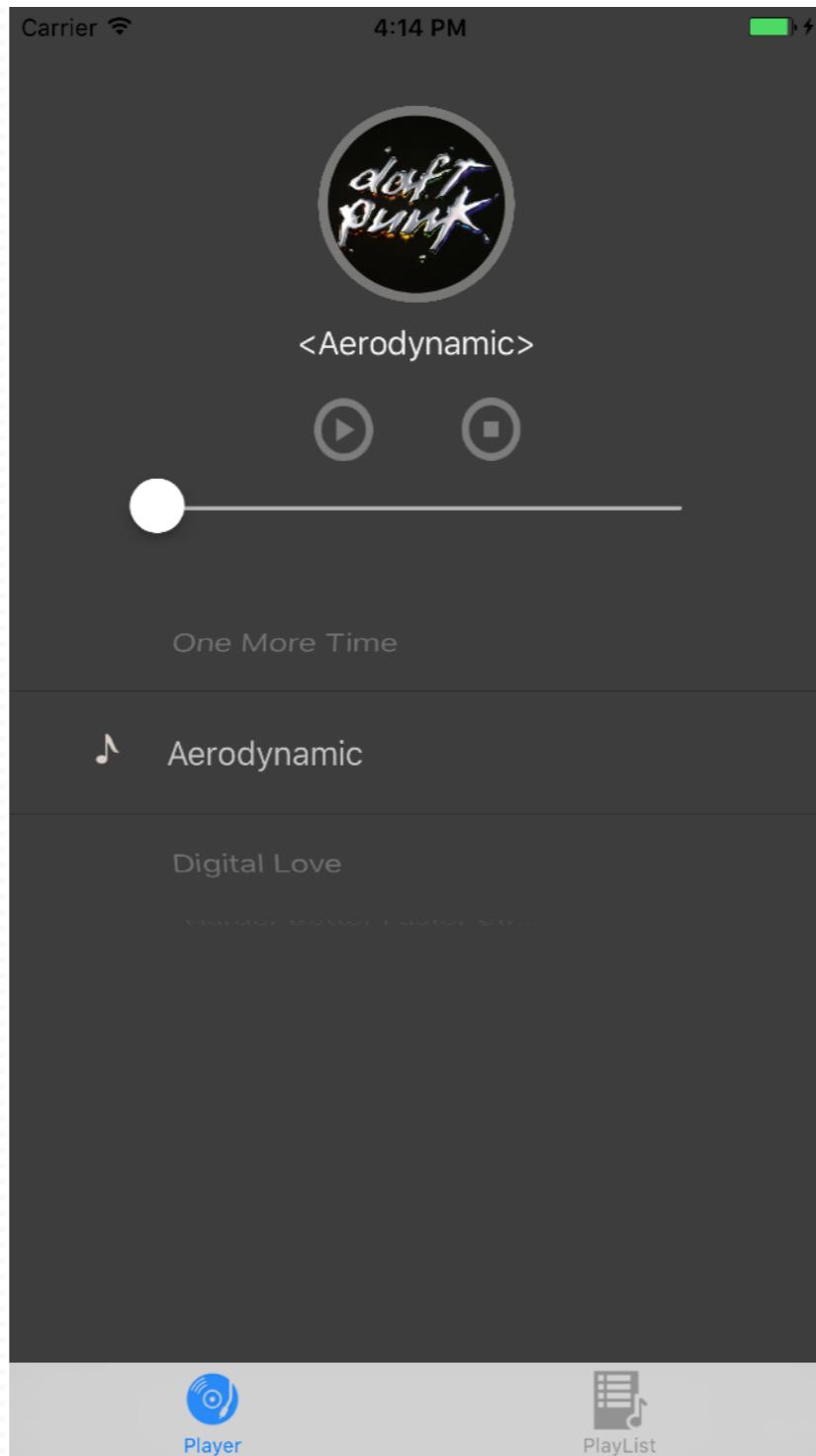
# Git

[https://github.com/bryanttang/iOS-  
Class-2016-9](https://github.com/bryanttang/iOS-Class-2016-9)

# Summary

- TabbarController
- Web view
- Access Network Data
- Play Audio

# Tab Bar Controller



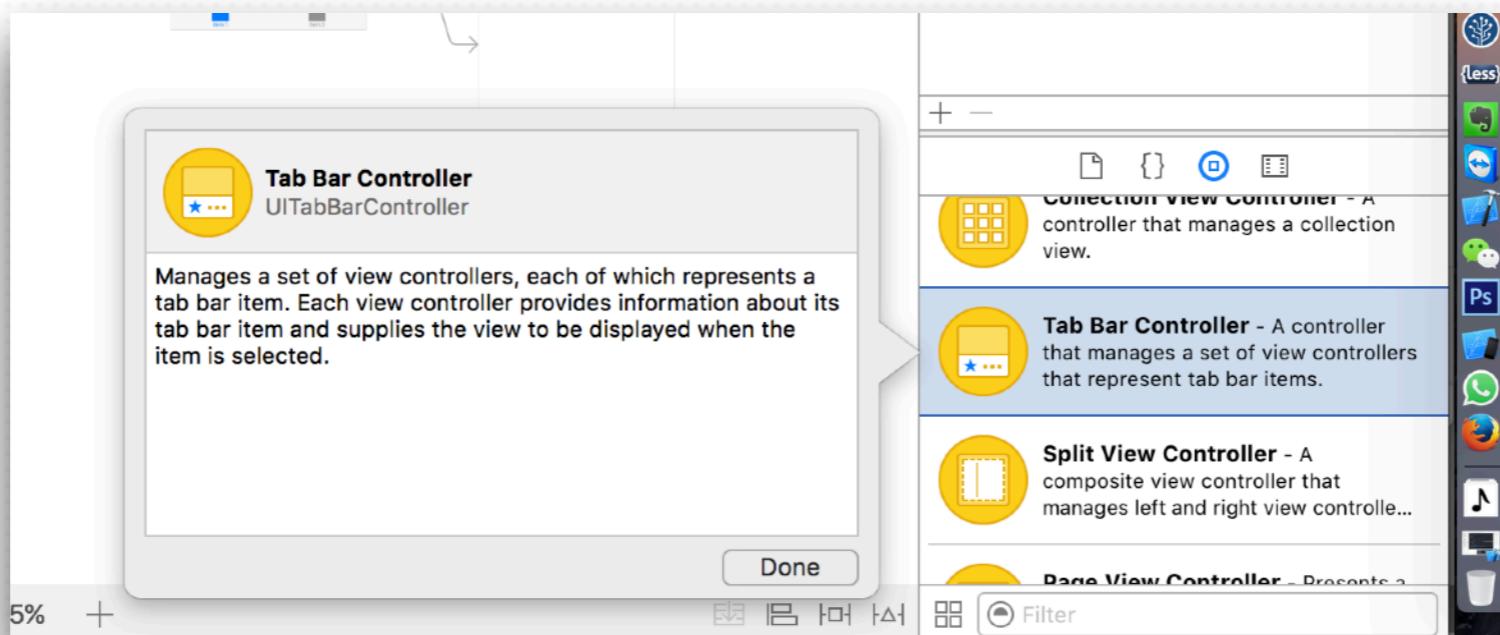
# Tab Bar Controller

- Two ways to create a Tabbarcontroller

- Use Tab Bar Controller object  in tool panel
- Embed in a viewcontroller

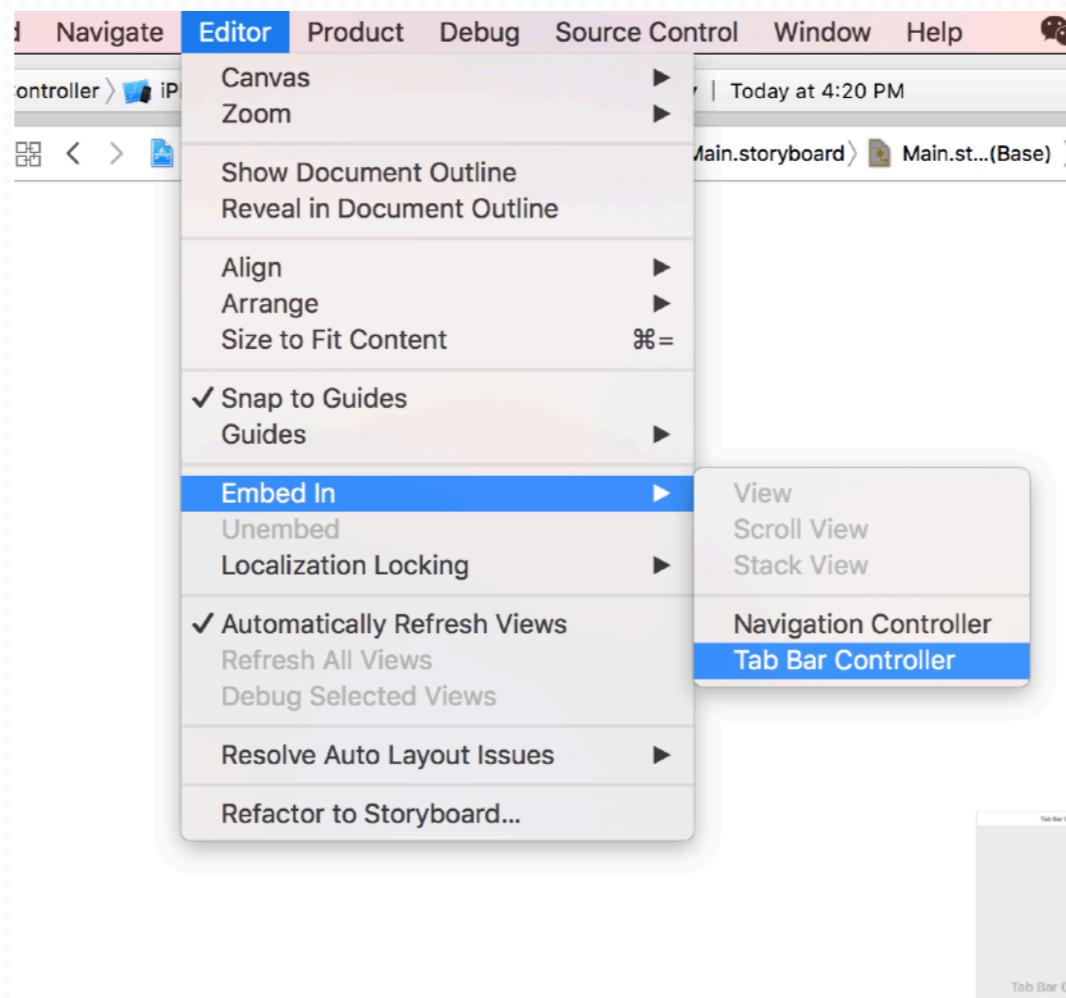
# Tab Bar Controller

- Use Tab Bar Controller object in tool panel



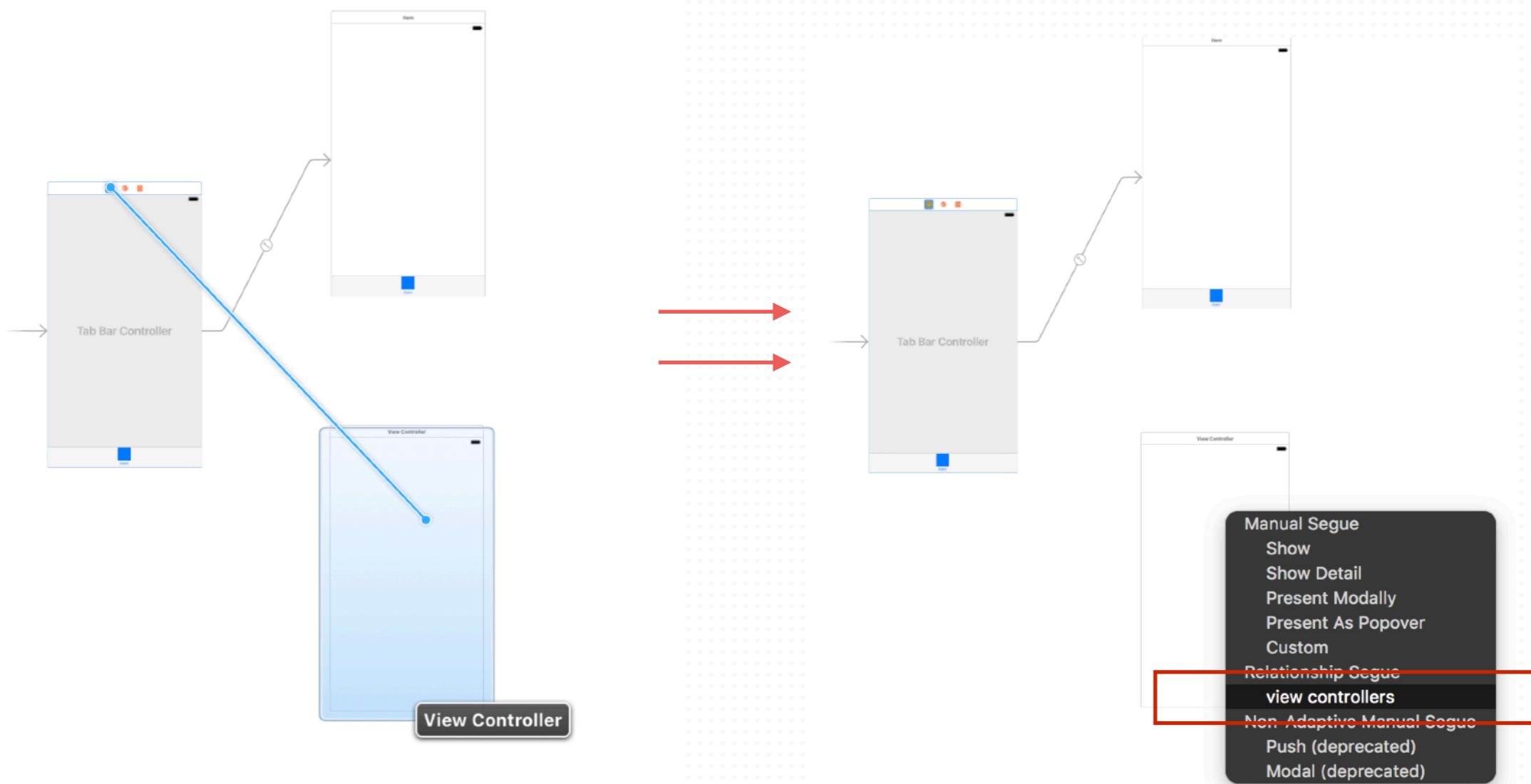
# Tab Bar Controller

- Embed in a viewcontroller



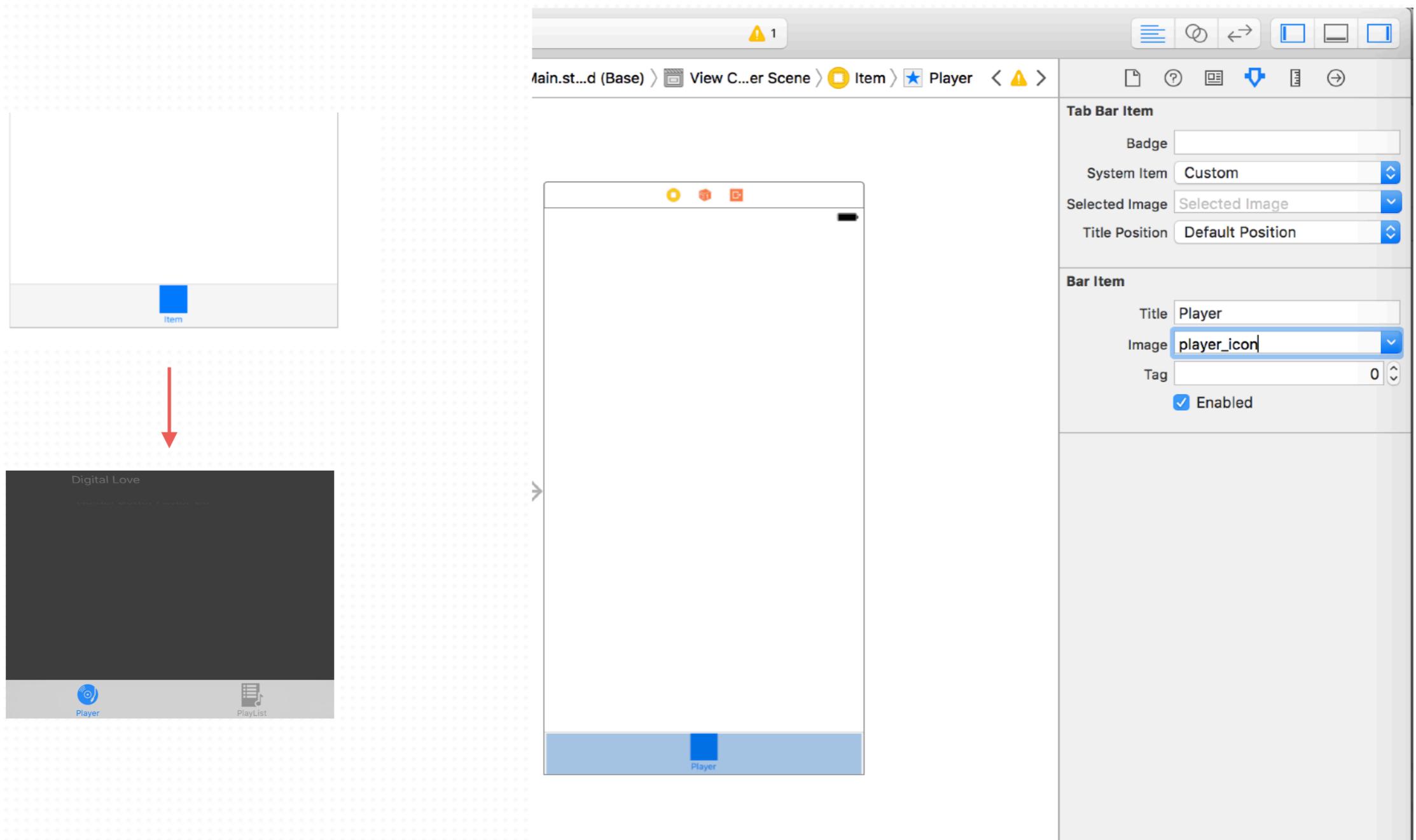
# Tab Bar Controller

- Add ViewController in Tab Bar Controller



# Tab Bar Controller

- Custom Tab bar icon



# Tab Bar Controller

- Change index in code

```
self.tabBarController.selectedIndex = 1;
```

# Demo

# Access Internet

# Get Data from URL

## ● App Transport Security [\(Must set before use\)](#)

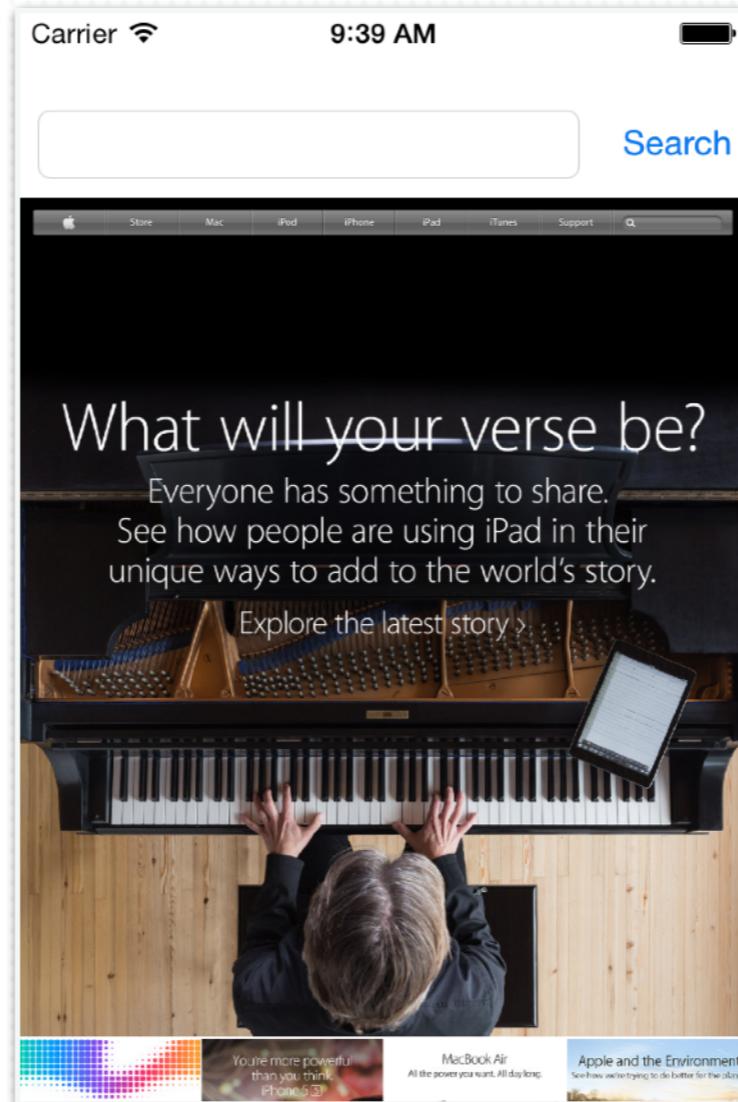
Adding the following to your Info.plist will disable ATS

```
01.  <key>NSAppTransportSecurity</key>
02.    <dict>
03.      <key>NSAllowsArbitraryLoads</key><true/>
04.    </dict>
```

Web view

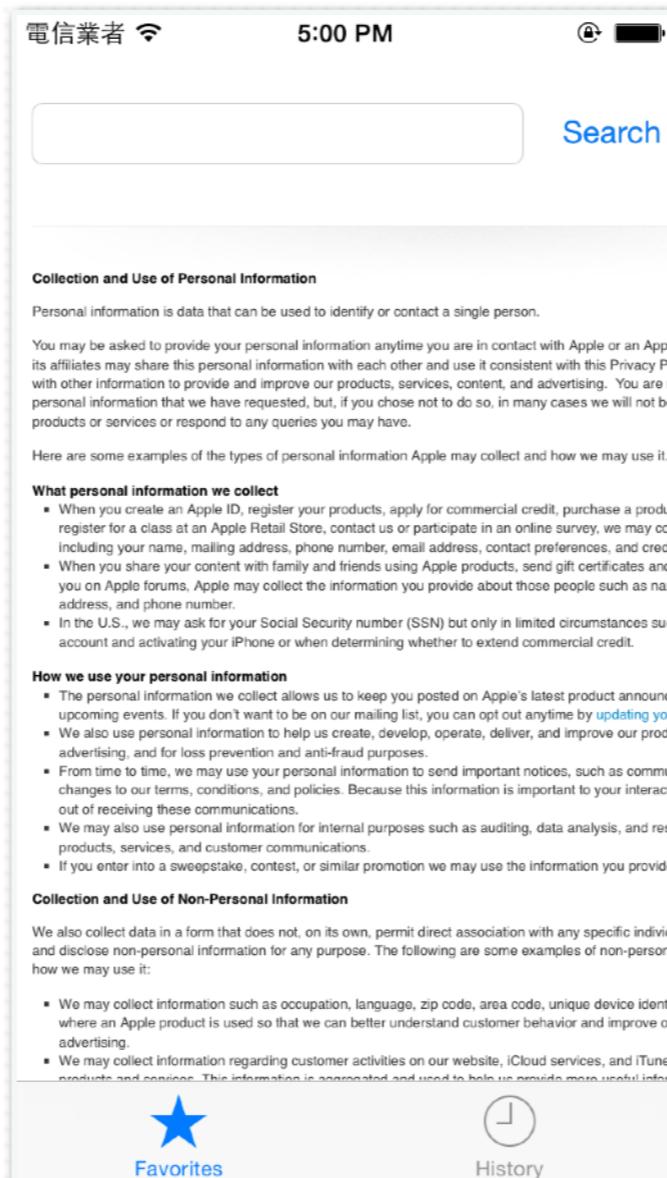
# Web view

- To show web content just like a browser



# Web view

- To show a page with static content



# Web view

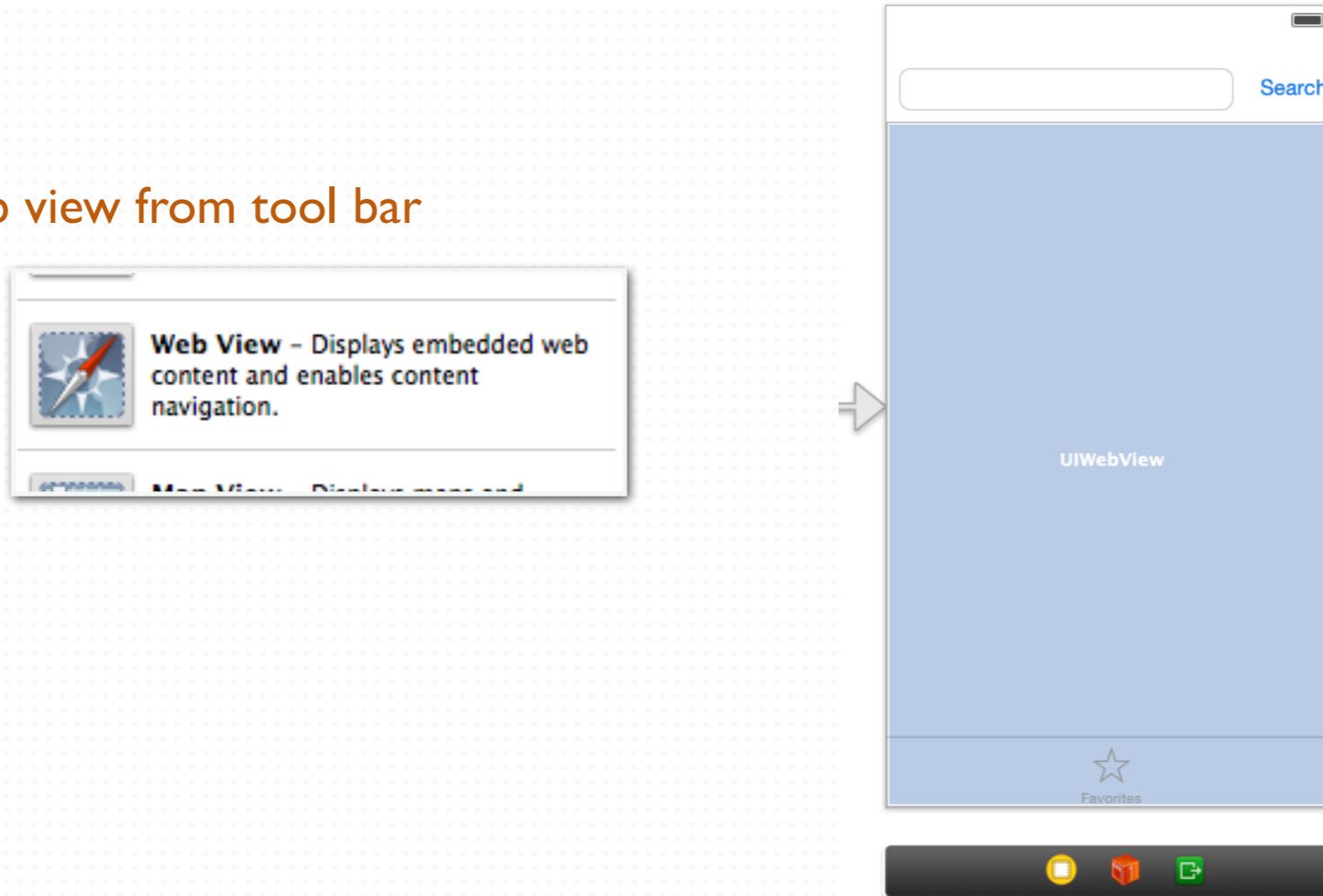
- To develop as a web app



# Web view

- Open a web view in project

Drag a web view from tool bar

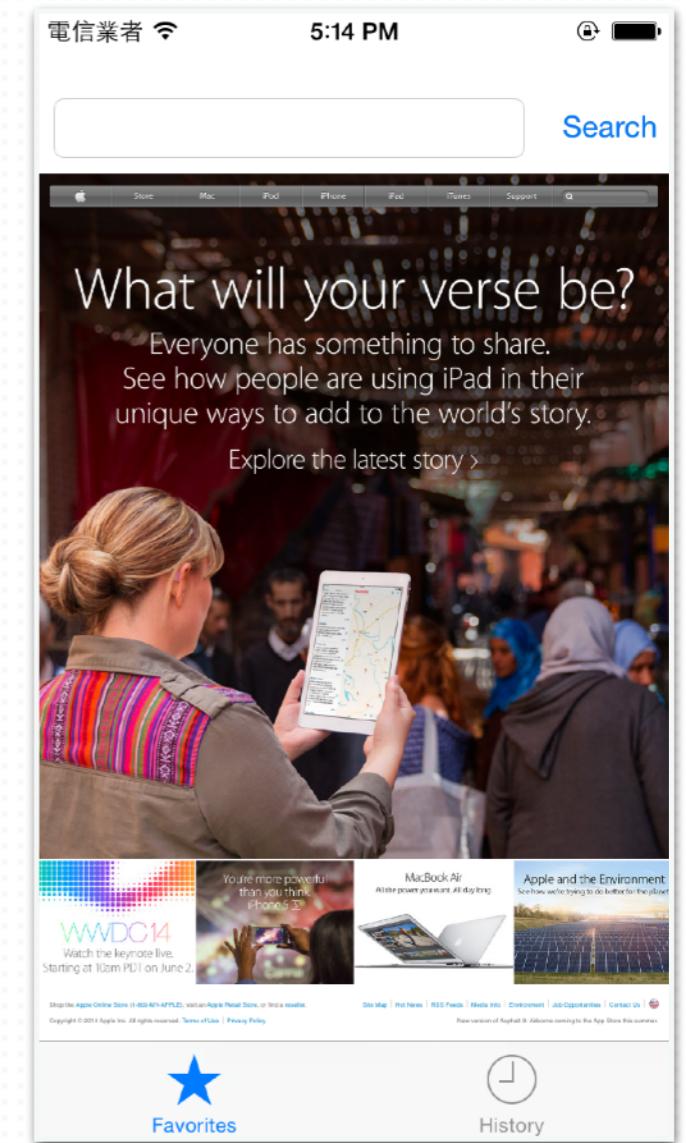


# Web view

- Load a website in web view

- create url string
- create request with the url
- load request in web view

```
NSURL *url = [NSURL URLWithString:@"http://www.apple.com"];
NSURLRequest *request = [NSURLRequest requestWithURL:url];
[_myWebView loadRequest:request];
[_myWebView setScalesPageToFit:YES]; //fit to page
```



# Get Data from URL

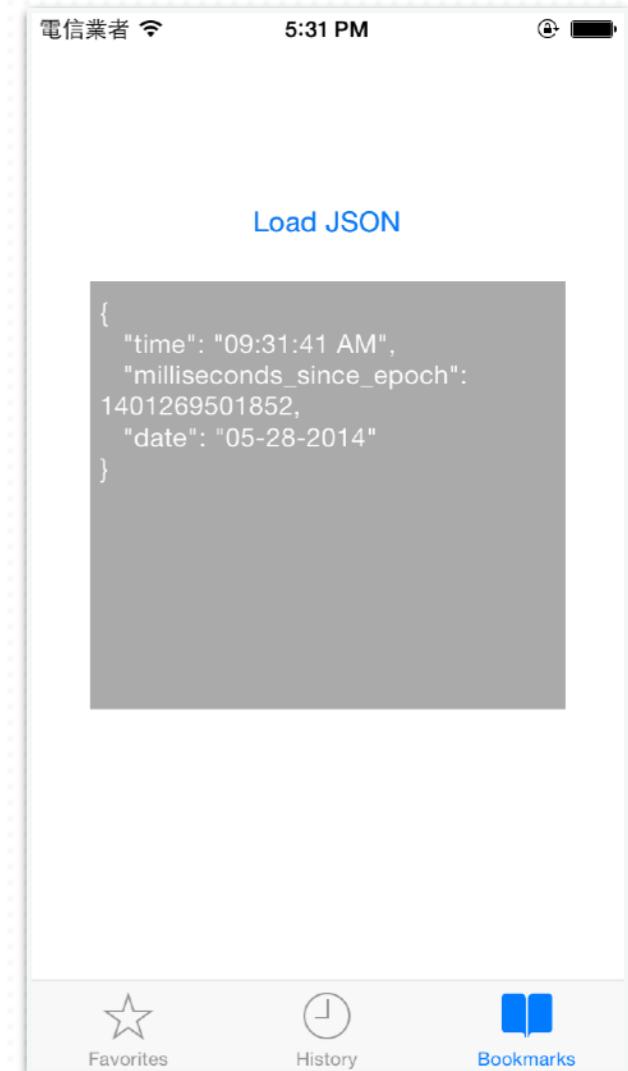
- Get String from URL
- Get Image from URL
- Get JSON, XML from URL

# Get Data from URL

- Load content by using **NSString**

- create url string
- Load url by using **NSString**

```
NSURL *url=[NSURL URLWithString:@"http://date.jsontest.com/"];  
  
NSString *json_str = [NSString stringWithContentsOfURL:url  
encoding:NSUTF8StringEncoding error:nil];
```

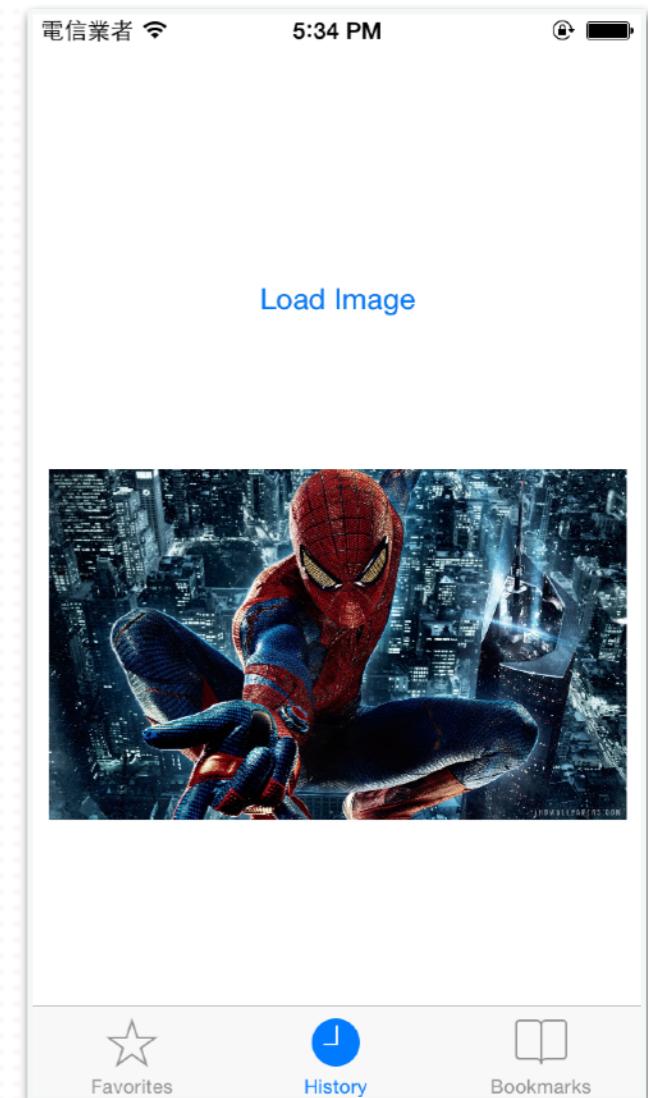


# Get Data from URL

- Load content by using UIImage

- create url string
- Load url by using NSData
- Convert NSData to UIImage

```
NSURL *url_image = [NSURL URLWithString:@"http://www.newyork-
business.com/wp-content/uploads/2014/02/
amazing_spider_man_latest-1920x1200.jpg"];  
  
NSData *image_data = [NSData dataWithContentsOfURL:url_image
options: NSDataReadingUncached error: nil];  
  
_imageView.image = [UIImage imageWithData:image_data];
```



# Get Data from URL

- Load content by using JSONSerialization
  - create url string
  - Load url by using NSData
  - Convert NSData to JSONSerialization Object

```
NSURL *url_json = [NSURL URLWithString:@"52.6.19.49/mission_cpttm/mission.php"];  
  
NSData *data_json = [NSData dataWithContentsOfURL:url_image  
options:NSDataReadingUncached error:nil];  
  
NSJSONSerialization *json = [NSJSONSerialization JSONObjectWithData:json_data  
options:NSJSONReadingAllowFragments error:nil];
```

# Play audio

- Two types of sound
  - Playing UI Sound Effects Using System Sound Services
  - Playing Sounds Easily with the AVAudioPlayer Class

# Play audio

- Playing UI Sound Effects Using System Sound Services
  - No longer than 30 seconds in duration
  - In linear PCM or IMA4 (IMA/ADPCM) format
  - Packaged in a .caf, .aif, or .wav file

# Play audio

## ● How to ?

1. Creating a sound ID object
2. Playing a system sound

### Sample code

```
@property CFURLRef  
soundFileURLRef;
```

```
@property SystemSoundID  
soundFileObject;
```

```
// Get the main bundle for the app  
CFBundleRef mainBundle = CFBundleGetMainBundle();
```

```
// Get the URL to the sound file to play. The file in this case  
// is "tap.aif"
```

```
_soundFileURLRef = CFBundleCopyResourceURL (  
    mainBundle,  
    CFSTR ("bullet_3"),  
    CFSTR ("wav"),  
    NULL  
);
```

```
// Create a system sound object representing the sound file  
AudioServicesCreateSystemSoundID (  
    _soundFileURLRef,  
    &_soundFileObject  
);
```

# Play audio

- Playing UI Sound Effects or Invoking Vibration Using System Sound Services
  - Play sounds of any duration
  - Play sounds from files or memory buffers
  - Loop sounds
  - Play multiple sounds simultaneously (although not with precise synchronization)
  - Control relative playback level for each sound you are playing
  - Seek to a particular point in a sound file, which supports application features such as fast forward and rewind
  - Obtain audio power data that you can use for audio level metering

# Play audio

- The AVAudioPlayer class lets you play sound in any audio format available in iOS, as described in [Table 1-1](#).

**Table 1-1** Audio playback formats and codecs

Audio decoder/playback format	Hardware-assisted decoding	Software-based decoding
AAC (MPEG-4 Advanced Audio Coding)	Yes	Yes, starting in iOS 3.0
ALAC (Apple Lossless)	Yes	Yes, starting in iOS 3.0
HE-AAC (MPEG-4 High Efficiency AAC)	Yes	-
iLBC (internet Low Bitrate Codec, another format for speech)	-	Yes
IMA4 (IMA/ADPCM)	-	Yes
Linear PCM (uncompressed, linear pulse-code modulation)	-	Yes
MP3 (MPEG-1 audio layer 3)	Yes	Yes, starting in iOS 3.0
$\mu$ -law and a-law	-	Yes

# Play audio

## ● How to ?

1. Define the sound path (location path or network path)
2. Create an AVPlayer object
3. Configure AVPlayer object

### Sample code

#### //Get Sound Path

```
NSString *soundFilePath = [[NSBundle mainBundle] pathForResource: @"bg_sound"  
                           ofType: @"wav"];
```

```
NSURL *fileURL = [[NSURL alloc] initFileURLWithPath: soundFilePath];
```

#### //Create an AVAudioPlayer

```
newPlayer = [[AVPlayer alloc] initWithURL: fileURL];
```

#### //Configuration

```
newPlayer.volume = 0.5;  
[newPlayer play];
```