

«PlayTrack»

Fabrice.Kordon@lip6.fr



Goal of the example

2

Operate the MediaPlayer framework

-  MPMediaPickerController
-  MPMusicPlayerController

Simple example

-  Access to the music library
-  Build a playlist
-  Play and navigate in this playlist

On the simulator?

-  No real interest (no music installed a priori)
-  Used to crash (it's better now 🎧)

Demo



ViewController



Sake of simplicity...

Code located in a
ViewController

ViewController

```
import UIKit
import MediaPlayer

class ViewController: UIViewController, MPMediaPickerControllerDelegate {

    private let myMediaPlayer = MPMusicPlayerController.applicationMusicPlayer

    private let multiple = UISwitch()
    private let b1 = UIButton(type: .system)
    private let b2 = UIButton(type: .system)
    private let b3 = UIButton(type: .system)
    private let b4 = UIButton(type: .system)
    private let l = UILabel()
```


ViewController

```
override func viewDidLoad() {
    super.viewDidLoad()
    self.view = UIView()
    self.view.backgroundColor = .white
    let legend = UILabel()
    legend.textAlignment = .left
    legend.text = "Multiple selection"
    self.view.addSubview(legend)
    multiple.isOn = true
    self.view.addSubview(multiple)
    b1.setTitle("Choose a track", for: .normal)
    self.view.addSubview(b1)
    b2.setTitle(">|", for: .normal)
    self.view.addSubview(b2)
    b3.setTitle("|<", for: .normal)
    self.view.addSubview(b3)
    b4.setTitle("stop", for: .normal)
    self.view.addSubview(b4)
    l.textAlignment = .center
    self.view.addSubview(l)
    b1.addTarget(self, action: #selector(chooseTrack), for: .touchDown)
    b2.addTarget(self, action: #selector(goNext), for: .touchDown)
    b3.addTarget(self, action: #selector(goPrevious), for: .touchDown)
    b4.addTarget(self, action: #selector(stop), for: .touchDown)
    // No orientation supported!
    let s = UIScreen.main.bounds.size
    multiple.frame = CGRect(x: 10, y: 50, width: 50, height: 25)
    legend.frame = CGRect(x: 70, y: 50, width: 150, height: 25)
    b1.frame = CGRect(x: s.width / 2 - 80, y: 100, width: 160, height: 25)
    b4.frame = CGRect(x: s.width / 2 - 30, y: 150, width: 60, height: 25)
    b3.frame = CGRect(x: s.width / 4 - 30, y: 200, width: 60, height: 25)
    b2.frame = CGRect(x: s.width / 4 * 3 - 30, y: 200, width: 60, height: 25)
    l.frame = CGRect(x: 10, y: 250, width: s.width - 20, height: 25)
}
```


ViewController

```
@objc func chooseTrack() {
    let myMediaPlayer = MPMediaPickerController(mediaTypes: MPMediaType.music)
    if multiple.isOn {
        myMediaPlayer.allowsPickingMultipleItems = true
        myMediaPlayer.prompt = "Please select tracks"
    } else {
        myMediaPlayer.prompt = "Please select a track"
    }
    myMediaPlayer.delegate = self
    myMediaPlayer.loadView()
    self.present(myMediaPlayer, animated: true, completion: nil)
}

@objc func goPrevious() {
    myMediaPlayer.skipToPreviousItem()
}

@objc func goNext() {
    myMediaPlayer.skipToNextItem()
}

@objc func stop() {
    myMediaPlayer.stop()
    l.text = "Stopped"
}
```


ViewController

```
@objc func updateTrackData() { // Service function
    let crTrack = myMediaPlayer.nowPlayingItem
    let a = crTrack?.artist
    let t = crTrack?.title
    l.text = "\(t ?? "no title") (\(a ?? "no artist"))"
}

// MPMediaPickerControllerDelegate protocol

func mediaPicker(_ mediaPicker: MPMediaPickerController,
                 didPickMediaItems mediaItemCollection: MPMediaItemCollection) {
    mediaPicker.dismiss(animated: true, completion: nil)
    myMediaPlayer.setQueue(with: mediaItemCollection)
    NotificationCenter.default.addObserver(self,
        selector: #selector(updateTrackData),
        name: NSNotification.Name.MPMusicPlayerControllerNowPlayingItemDidChange,
        object: nil)
    myMediaPlayer.play()
    myMediaPlayer.beginGeneratingPlaybackNotifications()
}

func mediaPickerDidCancel(_ mediaPicker: MPMediaPickerController) {
    mediaPicker.dismiss(animated: true, completion: nil)
    myMediaPlayer.stop()
}
}
```


info.plist

```
21 <key>LSRequiresiPhoneUS</key>
22 <true/>
23 <key>UILaunchStoryboardName</key>
24 <string>LaunchScreen</string>
25 <key>UIMainStoryboardFile</key>
26 <string>Main</string>
27 <key>UIRequiredDeviceCapabilities</key>
28 <array>
29   <string>armv7</string>
30 </array>
31 <key>UISupportedInterfaceOrientations</key>
32 <array>
33   <string>UIInterfaceOrientationPortrait</string>
34 </array>
35 <key>UISupportedInterfaceOrientations~ipad</key>
36 <array>
37   <string>UIInterfaceOrientationPortrait</string>
38 </array>
39 <key>NSAppleMusicUsageDescription</key>
40 <string>It is better for a player to access you music ;-)</string>
41 </dict>
42 </plist>
43
```

Do not forget it!
If you want it to work...

As a conclusion...



Easy is'n't it?



Remember, iCloud access is «free»...
... for those who trust cloud

