

## sonidos y botones simples

ahora es posible sonar sonidos de alerta de un mod muy simple en iOS, usando la biblioteca <AudioToolbox/AudioToolbox.h>

pero este tipo de sonidos desde el sistema solo tener una duración menor a los 30 segundos y sonar pocos formatos de audio como mp3, aif, wav

aun así para muchas aplicaciones como en juegos por ejemplo son mucho más fáciles de usar que los modos tradicionales.

lo primero es importar la biblioteca y declarar la variable que se usará para albergar el sonido mismo, el tipo "SystemSoundID" es interno a la biblioteca así que se puede usar inmediatamente después de haber importado la misma, esto se hace en el .h del controlador principal:

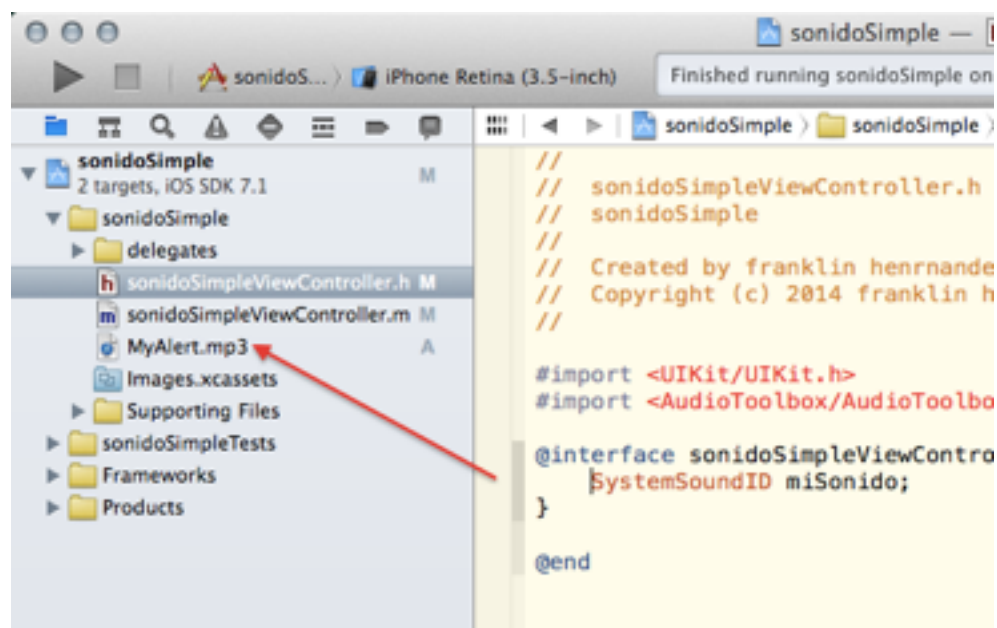
```
//
// sonidoSimpleViewController.h
// sonidoSimple
//
// Created by franklin henrnandez-castro on 25/7/14.
// Copyright (c) 2014 franklin henrnandez-castro. All rights reserved.
//

#import <UIKit/UIKit.h>
#import <AudioToolbox/AudioToolbox.h>

@interface sonidoSimpleViewController : UIViewController{
    SystemSoundID miSonido;
}

@end
```

el siguiente paso es importar el sonido para eso solo hay que arrastrar un sonido de menos de 30 segundos en un formato compatible, en nuestro caso usamos un mp3:



el siguiente paso es hacer los cambios necesarios en el .m del controlador principal. lo primero es declarar la dirección donde está el sonido y asociarlo con la variable que declaramos antes, esto se hace en alguna de las funciones de setup, en nuestro caso yo lo hice en el viewDidLoad:

```
#import "sonidoSimpleViewController.h"

@interface sonidoSimpleViewController ()
@end

@implementation sonidoSimpleViewController

- (void)viewDidLoad {
    [super viewDidLoad];
    float anchoScreen = [self.view bounds].size.width;

    // carga del sonido
    NSURL *soundURL = [[NSBundle mainBundle] URLForResource:@"MyAlert"withExtension:@"mp3"];
    AudioServicesCreateSystemSoundID((__bridge CFURLRef)soundURL, &miSonido);

    // boton
    UIButton *botonEscenario1 = [UIButton buttonWithType:UIButtonTypeRoundedRect];
    botonEscenario1.frame = CGRectMake(anchoScreen/5, 200, 200, 50);
    UIFont *myFont = [UIFont fontWithName:@"HelveticaNeue-Light" size:20.0f];
    UIFont *systemFont = myFont;
    botonEscenario1.titleLabel.font = systemFont;
    botonEscenario1.tag=0;
    botonEscenario1.titleLabel.textAlignment = NSTextAlignmentCenter;
    botonEscenario1.tintColor = [UIColor darkGrayColor];
    [botonEscenario1 setTitle:@"tocar sonido" forState:UIControlStateNormal];
    [botonEscenario1 addTarget:self action:@selector(wasTapped:withEvent:)forControlEvents:
    UIControlEventTouchUpInside];
    [self.view addSubview:botonEscenario1];
}

```

con esto el sonido ya está asociado a la variable y lo único que falta es decidir cuándo va a sonar, en nuestro caso se definió un botón que se creó desde el código:

```
#import "sonidoSimpleViewController.h"

@interface sonidoSimpleViewController ()
@end

@implementation sonidoSimpleViewController

- (void)viewDidLoad {
    [super viewDidLoad];
    float anchoScreen = [self.view bounds].size.width;

    // carga del sonido
    NSURL *soundURL = [[NSBundle mainBundle] URLForResource:@"MyAlert"withExtension:@"mp3"];
    AudioServicesCreateSystemSoundID((__bridge CFURLRef)soundURL, &miSonido);

    // boton
    UIButton *botonEscenario1 = [UIButton buttonWithType:UIButtonTypeRoundedRect];
    botonEscenario1.frame = CGRectMake(anchoScreen/5, 200, 200, 50);
    UIFont *myFont = [UIFont fontWithName:@"HelveticaNeue-Light" size:20.0f];
    UIFont *systemFont = myFont;
    botonEscenario1.titleLabel.font = systemFont;
    botonEscenario1.tag=0;
    botonEscenario1.titleLabel.textAlignment = NSTextAlignmentCenter;
    botonEscenario1.tintColor = [UIColor darkGrayColor];
    [botonEscenario1 setTitle:@"tocar sonido" forState:UIControlStateNormal];
    [botonEscenario1 addTarget:self action:@selector(wasTapped:withEvent:)forControlEvents:
    UIControlEventTouchUpInside];
    [self.view addSubview:botonEscenario1];
}

```

finalmente hay que implementar la función wasTapped que se menciona es la inicialización del botón:

```
- (void)wasTapped:(UIButton *)button withEvent:(UIEvent *)event {
    // para entrar en el escenario 1
    if (button.tag==0) {
        AudioServicesPlayAlertSound(miSonido);
    }
}

```

nótese que se usa la declaración del tag (en este caso 0) para poder administrar en la misma función varios botones en caso de que sea necesario