

# MOV02 - Laboratorio 4

Juan Pablo Arias Mora

Febrero 2021

## 1 Introducción

El laboratorio anterior nos mostró como realizar transiciones entre **View Controllers** utilizando solamente **Storyboards**. En este caso realizaremos una transiciones utilizando solamente código.

Para referencias futuras el código fuente final del mismo esta disponible en:  
<https://github.com/pabloariasora/MOV02-Cenfotec-Demo-Labs.git>

### 1.1 Versiones

- MacOS 10.15.7 - Inglés
- Xcode 12.4
- Simulador iPhone 11 Pro Max

## 2 Instrucciones

'Todo el mundo aquí tiene la sensación de que ahora es uno de esos momentos que están influenciando el futuro' Steve Jobs

### 2.1 Creación de aplicación base

Paso 1: Cree un nuevo proyecto en Xcode, utilizando el template App dentro de iOS, en la ventana de opciones para la configuración especifica de la nueva aplicación.

**Product Name** : transitions  
**Organization Identifier** : com.cenfotec.mov02.04  
**Interface** : StoryBoard  
**Life Cycle** : UIKit App Delegate  
**Language** : Swift  
**Use Core Data**: Sin Seleccionar  
**Include Test**: Sin Seleccionar

### 2.2 Creación de un Storyboard base

Paso 1: Abrir el archivo **Main.storyboard**. Y utilizando **Object Library** agregue un segundo **View Controller**. Con el **Attribute Inspector** cambie el color de fondo de ambos utilizando la propiedad **Background**, uno por **System Orange** y el otro por **System Green**. (ver figura 1).

Paso 2: En **View Controller** naranja coloque un **Button** con **Constraints - Horizontal in Container** y **Vertical in Container**, y usando el **Attribute Inspector** cambie su **Text** por **Next**, y su **Font** a 40. (ver figura 2).

Paso 3: Debemos de colocar el View Controller naranja dentro de un **Navigation Controller**. Esto para poder usar las transiciones siguientes, esto se logra seleccionado el objeto **Table View Controller** en **Document Outline**, luego en la barra de de menú de Xcode seleccionamos **Editor - Embed In - Navigation Controller**. (visto en laboratorio 3).

Paso 4: En **View Controller** verde coloque un **Button** con **Constraints - Horizontal in Container** y **Vertical in Container**, y usando el **Attribute Inspector** cambie su **Text** por **Dismiss**, y su **Font** a 40. (ver figura 2).

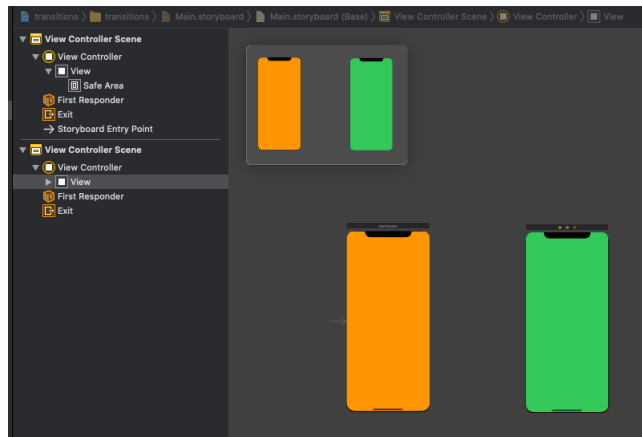


Figure 1: Vista Inicial del Proyecto

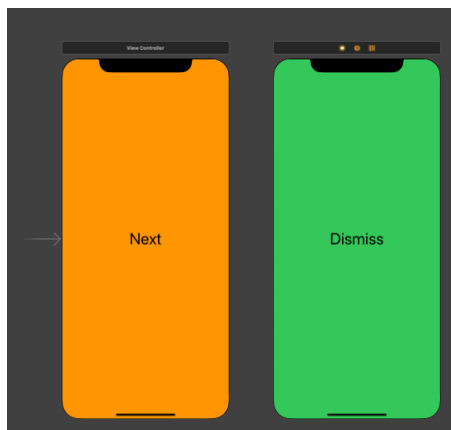


Figure 2: Vista Inicial del Proyecto con Botones

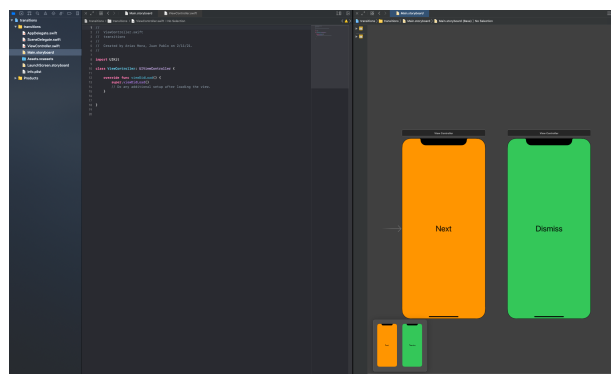


Figure 3: Editor Multiarchivo

- Paso 5: Abrir el archivo **ViewController.swift**. En una segunda ventana dentro del **Editor**, preferiblemente al lado izquierdo del **Main.storyboard**. (ver figura 3)
- Paso 6: Usando el **Document Outline** seleccione el **View Controller** verde, y dentro de las opciones **Identity Inspector**, busque las opciones **Identity** y dentro de esta la propiedad **Storyboard ID**, coloque el valor de **GreenViewController**.
- Paso 7: Usando el **Document Outline** seleccione el **View Controller** naranja, y dentro de las opciones **Identity Inspector**, busque las opciones **Identity** y dentro de esta la propiedad **Storyboard ID**, coloque el valor de **OrangeViewController**.
- Paso 8: Creamos un archivo nuevo de tipo **Swift File**, por nombre colocamos **GreenViewController** y dentro del mismo agregamos el siguiente código:

```
import Foundation
import UIKit

class GreenViewController: UIViewController {

    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading the view.
    }

}
```

Este código simplemente está definiendo una clase de herencia de **UIViewController**, su código es igual al código inicial dentro de **ViewController.swift**

Paso 9: Usando el **Document Outline** seleccione el **View Controller** verde, y dentro de las opciones **Identity Inspector**, busque las opciones **Custom Class** y dentro de esta la propiedad **Class**, coloque el valor de **GreenViewController**. Esto asociará el código que generamos con el **ViewController** del **Storyboard**.

Paso 10: **Control-Drag** el botón **Next** hacia el archivo **ViewController.swift**. (ver figura 4), y dentro de las opciones del menú colocar **Name** el valor **SegueTapped**.

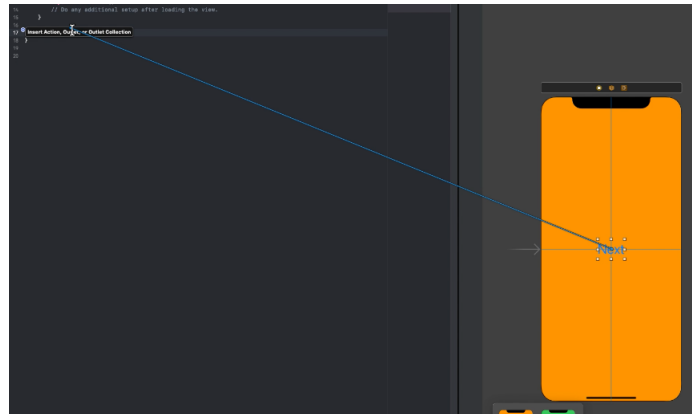


Figure 4: Control-Drag

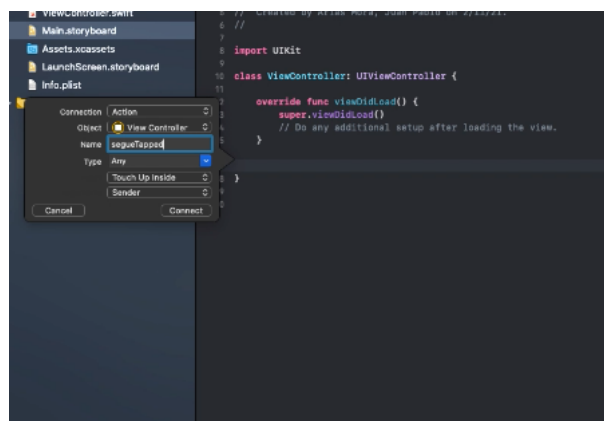


Figure 5: Asignación de Acción de Storyboard a View Controller

Paso 11: El paso anterior debió generarnos una función de tipo **IBAction** con nombre **SegueTapped**. dentro de la cual agregamos el siguiente código:

```
let mainStoryboard = UIStoryboard(name: "Main", bundle: Bundle.main)

guard let greenViewController = mainStoryboard.instantiateViewController(
```

```

withIdentifier: "GreenViewController") as?
    GreenViewController else{
        print("Couldn't find the view controller")
        return
    }

//// This is due to the relation that we declare using Editor-Embedded In-Navigation View Controller
//// Comment Below line in Step 13
navigationController?.pushViewController(greenViewController, animated: true)

//// Uncomment Below line in Step 13
//present(greenViewController, animated: true, completion: nil)
////Above line present "similar" functionality asnavigationController?.pushViewController
////But does not have a Navigation Stack therefore we don't have the back button

```

Como explicación instanciamos el **Storyboard** cuyo nombre es **Main**, para luego dentro del mismo hacer una búsqueda por el **StoryboardID GreenViewController**, de este no estar presente mostrar un error en la consola de **Debug**.

Para luego utilizar el **Navigation Controller** y mostrar el **View Controller** verde. Este objeto esta presente ya que realizamos el proceso de **Embedded In**

Paso 12: Compile y ejecute la aplicación. Presione sobre el botón **Next** y vea como se realiza la transición. Y el **Navigation Controller** nos habilita un botón **Back** para regresar.(ver figura 6)

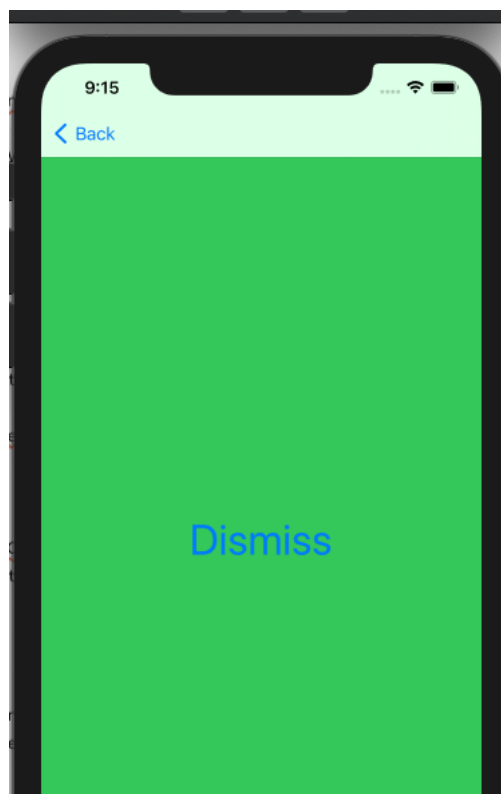


Figure 6: View Controller Verde con boton de Back

Paso 13: El hecho de tener un botón de **Back**, es debido a que dentro de nuestro **Navigation Controller** agregamos un camino de Navegación dentro del **Navigation Stack**, la forma de realizar un acercamiento similar por medio de código se encuentra dentro de los comentarios al final del código antes copiado, por tanto comentamos la línea que inicia por **navigationController** y des-comentamos solamente la línea que inicia por **present**

Paso 14: Compile y ejecute la aplicación. Presione sobre el botón **Next** y vea como se realiza la transición. Vemos como

esta vez carece del botón de **Back** para regresar, solamente que ahora para regresar debemos hacer hacia abajo el **View Controller** verde. (ver figura 7)

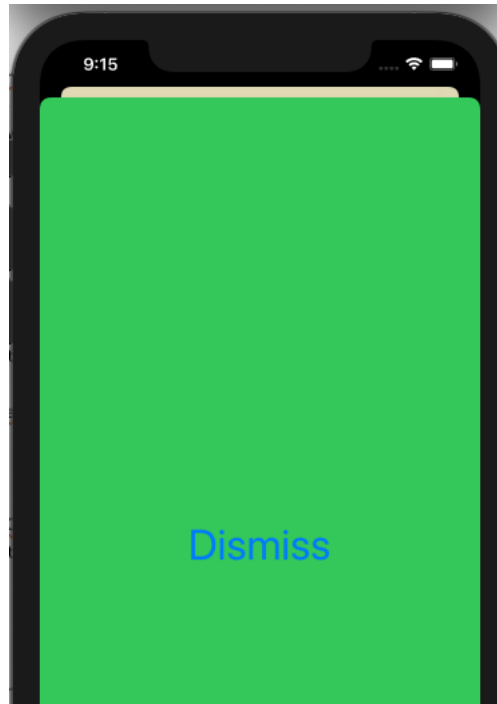


Figure 7: View Controller Verde con boton de Back

Paso 15: Dentro del archivo **GreenViewController.swift** agregamos el siguiente código:

```
@IBAction func dismissTapped(_ sender: Any) {  
    dismiss(animated: true, completion:nil)  
}
```

Paso 16: Realizamos **Control-Drag** el botón **Dismiss** hacia el archivo **GreenViewController.swift**, justo en la definición de la función **dismissTapped** (ver laboratorio 2).

Paso 17: Compile y ejecute la aplicación. Presione sobre el botón **Next** y vea como se realiza la transición. Presione sobre el botón **Dismiss** y vea como se realiza la transición.

Paso 18: Si deseamos modificar la animación, dentro del archivo **ViewController.swift**, justo previo a la llamada de la función **present(greenViewController, animated: true, completion: nil)**, podemos agregar el siguiente código:

```
greenViewController.modalTransitionStyle = .coverVertical
```

Paso 19: Compile y ejecute la aplicación. Presione sobre el botón **Next** y vea como se realiza la transición.

Paso 20: Repita los cambios de transiciones para experimentar con los valores de **.crossDissolve** - **.flipHorizontal**