

APLICACIÓN CON TAB BAR

.....
*Programación de Dispositivos
Móviles*

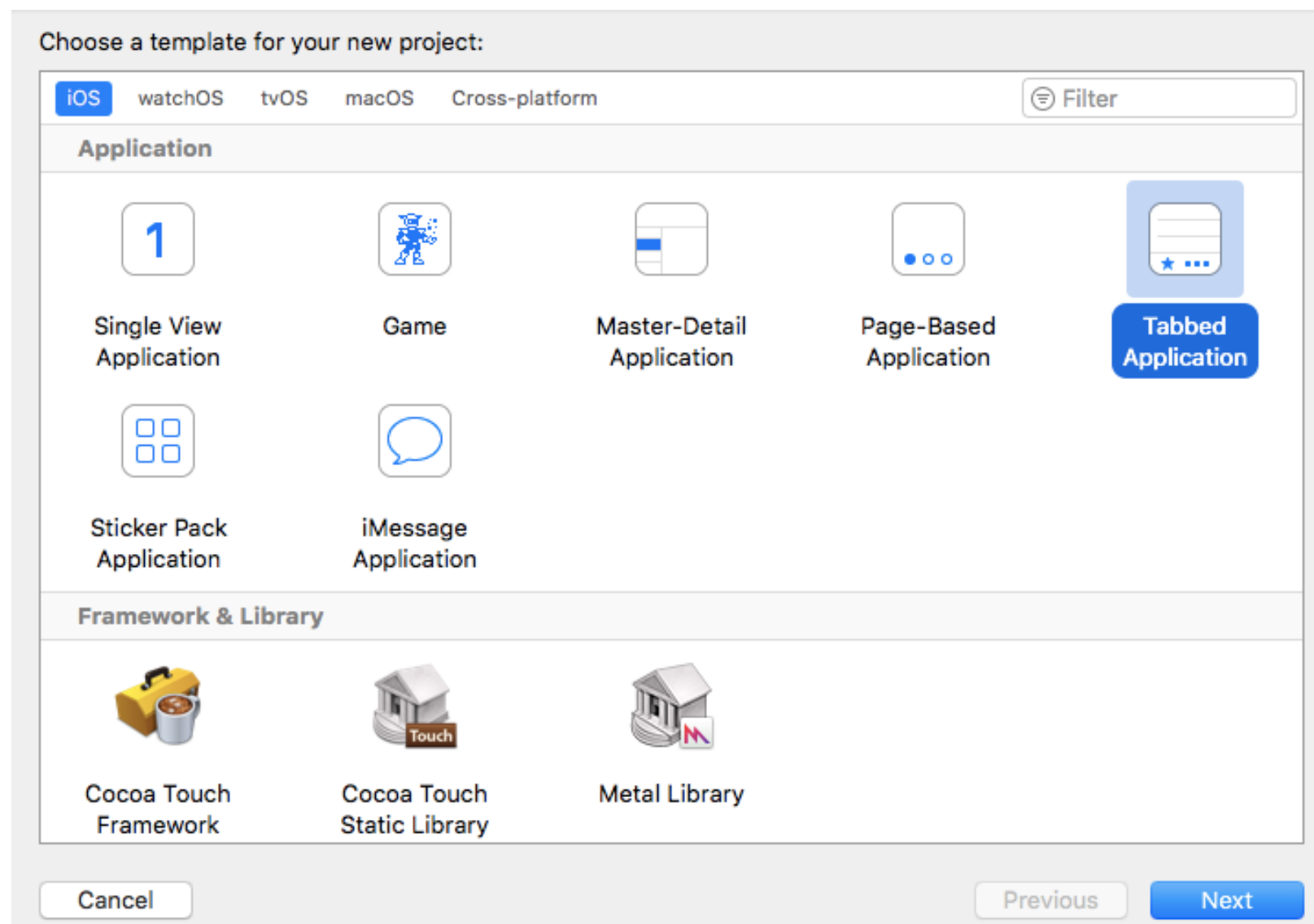


OBJETIVO

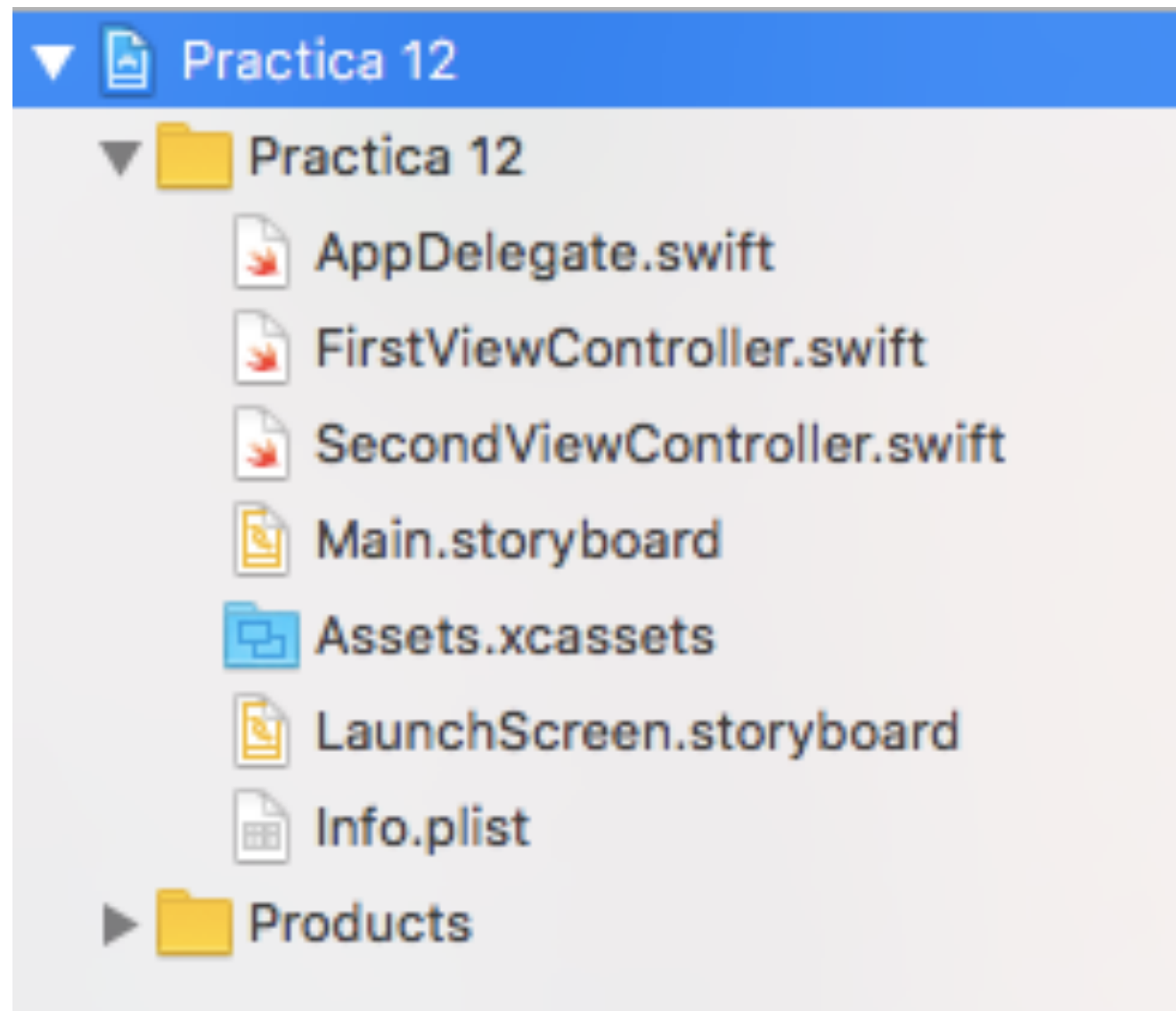
- Crear una aplicación de tres vistas usando el componente UITabBarController.

INICIANDO

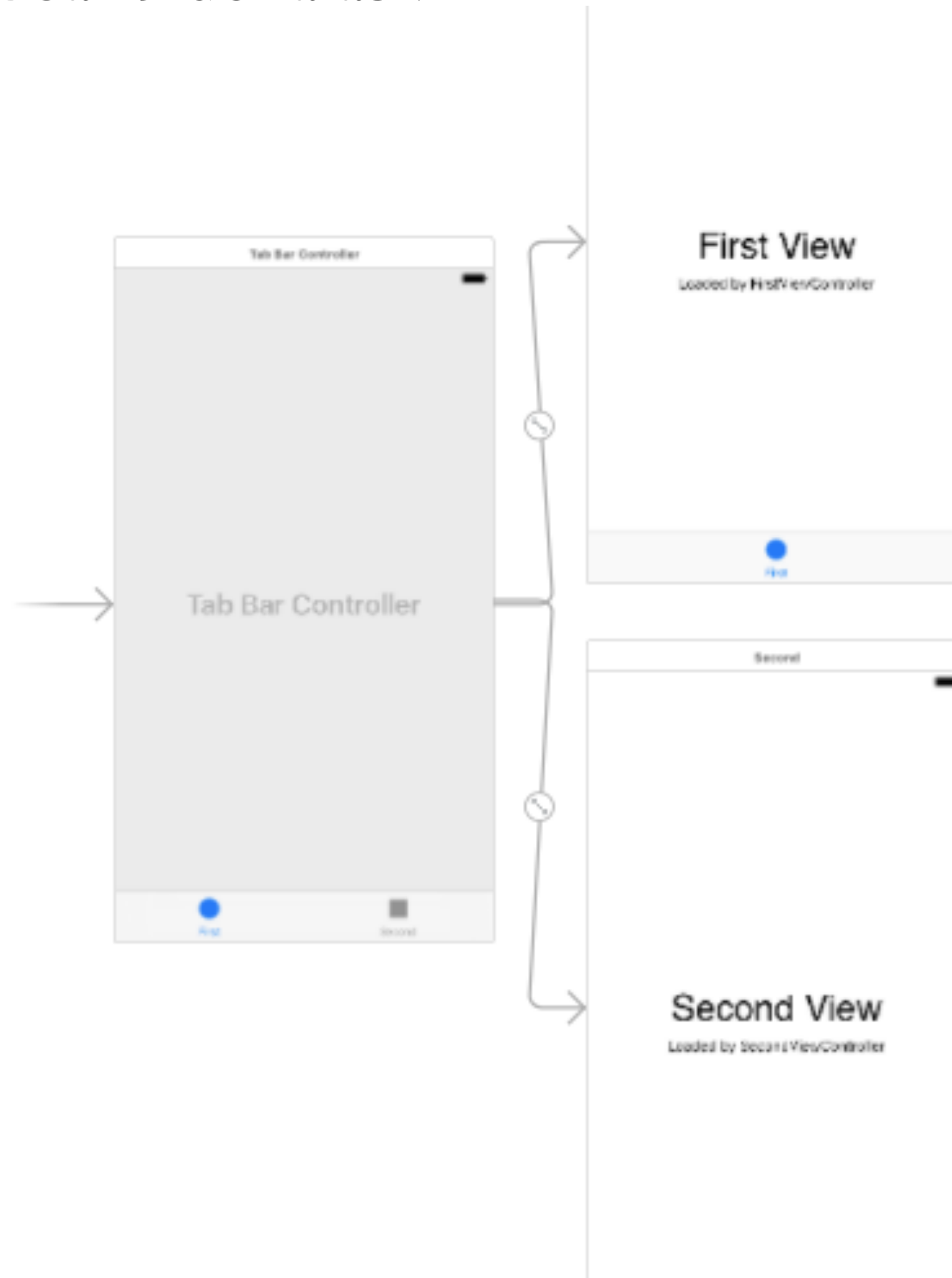
- Crea un nuevo proyecto, pero esta vez en lugar de seleccionar Single View Application, selecciona Tabbed Application.



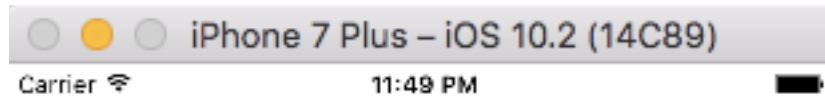
- Como podrás observar se crearon dos nuevos archivos llamados FirstViewController.swift y SecondViewController.swift



- Y el storyboard lucirá así:

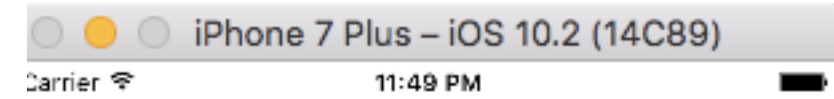
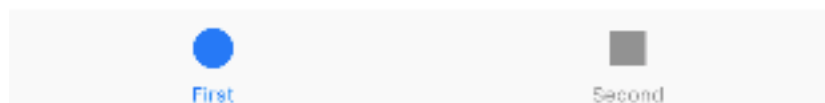


► Ejecuta tu aplicación:



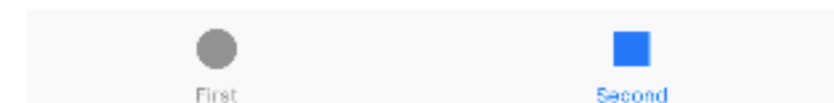
First View

Loaded by FirstViewController



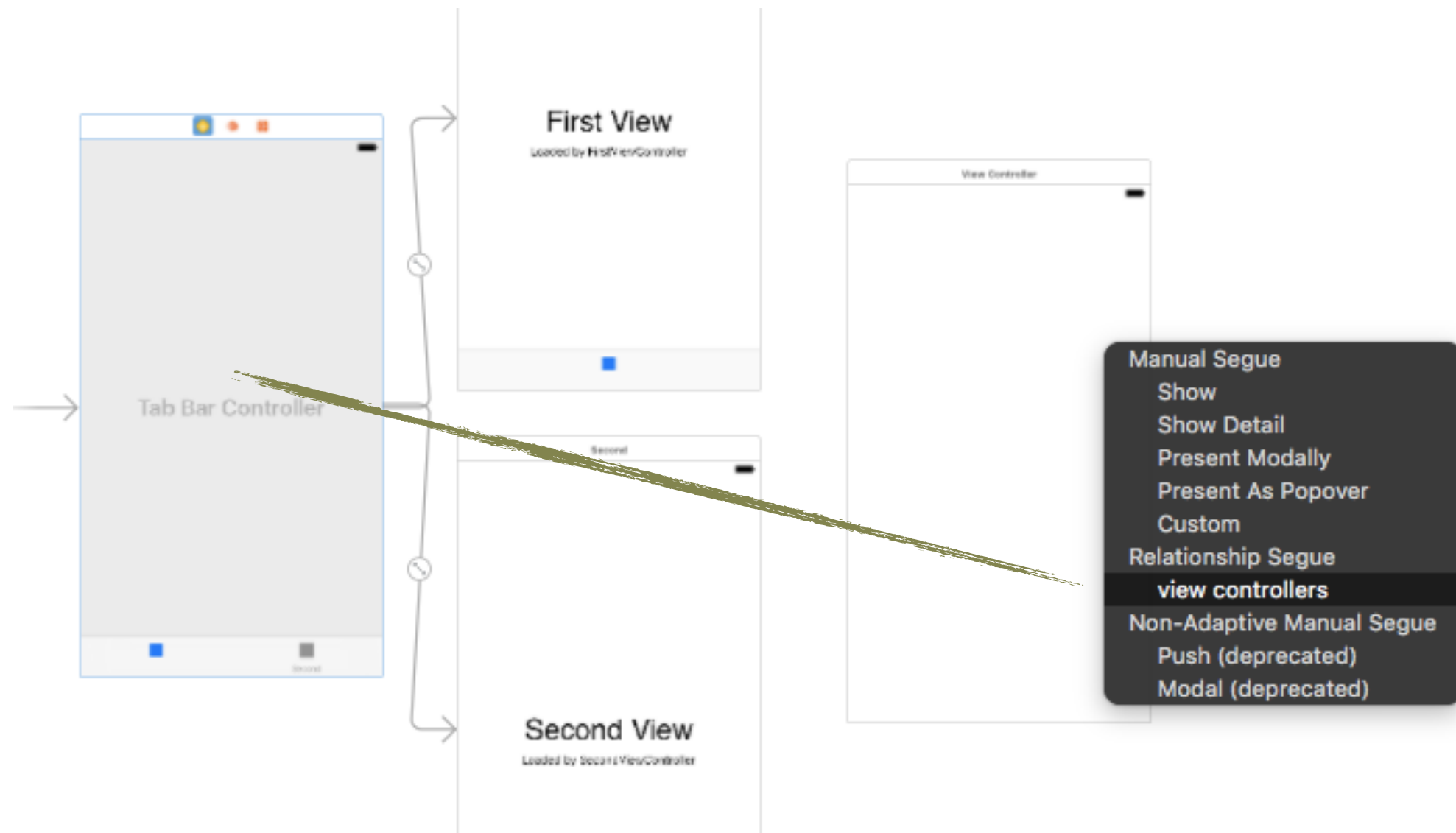
Second View

Loaded by SecondViewController



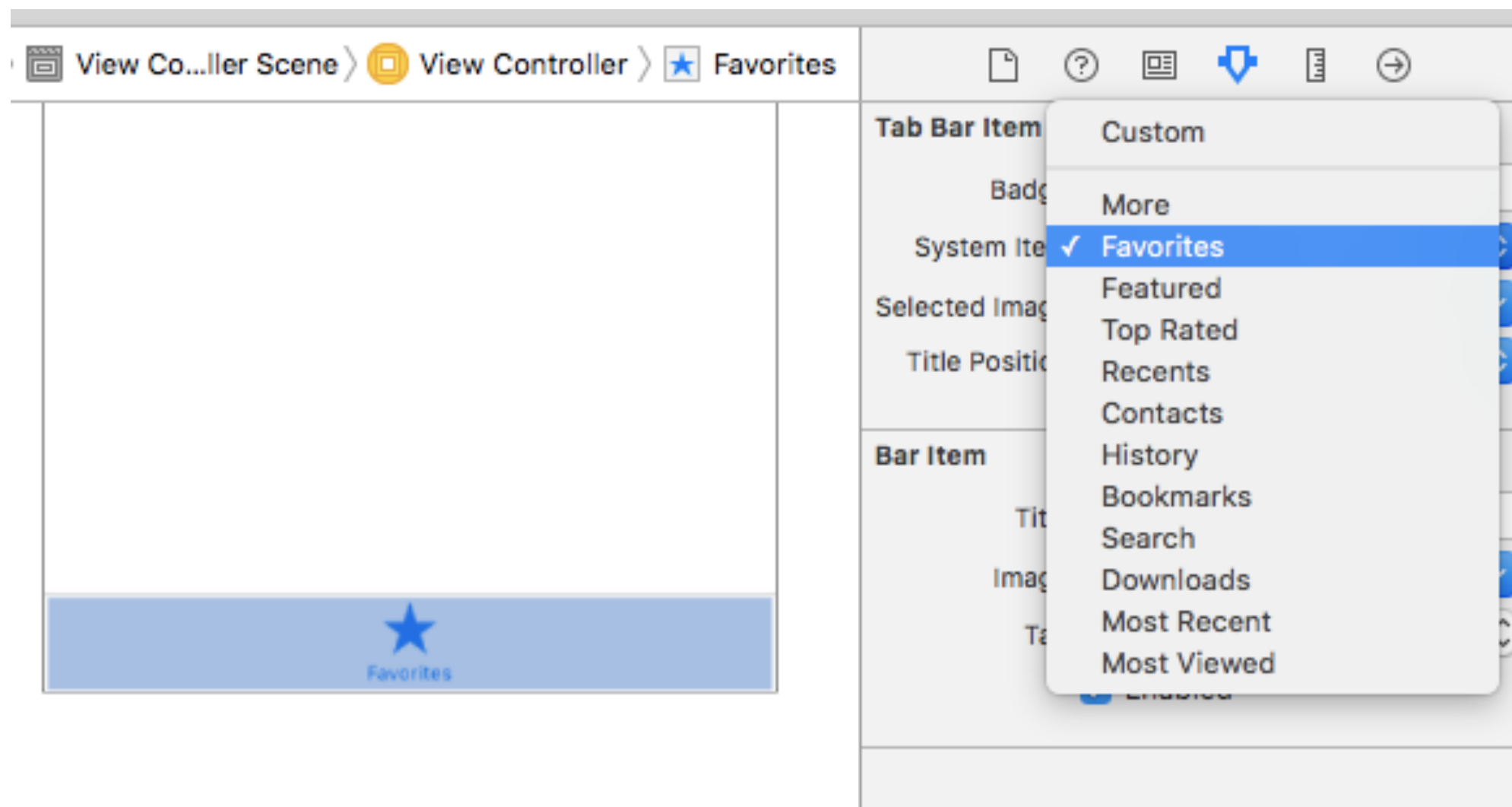
AGREGANDO OTRO TAB BAR

- Arrastra un componente ViewController.
- Para vincular la nueva vista da clic en el Tab Bar Controller, presiona control y arrastra hasta la nueva vista, selecciona **view controllers**.

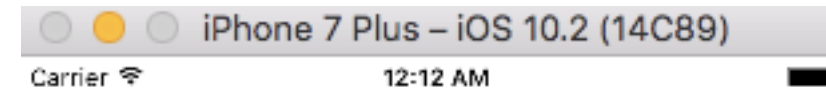


MODIFICANDO EL ICONO

- Para modificar el icono del tab bar, da clic sobre la barra inferior de la vista y en la opción System ítem, selecciona Favorites.



- Ahora tu aplicación debe lucir así:



First View

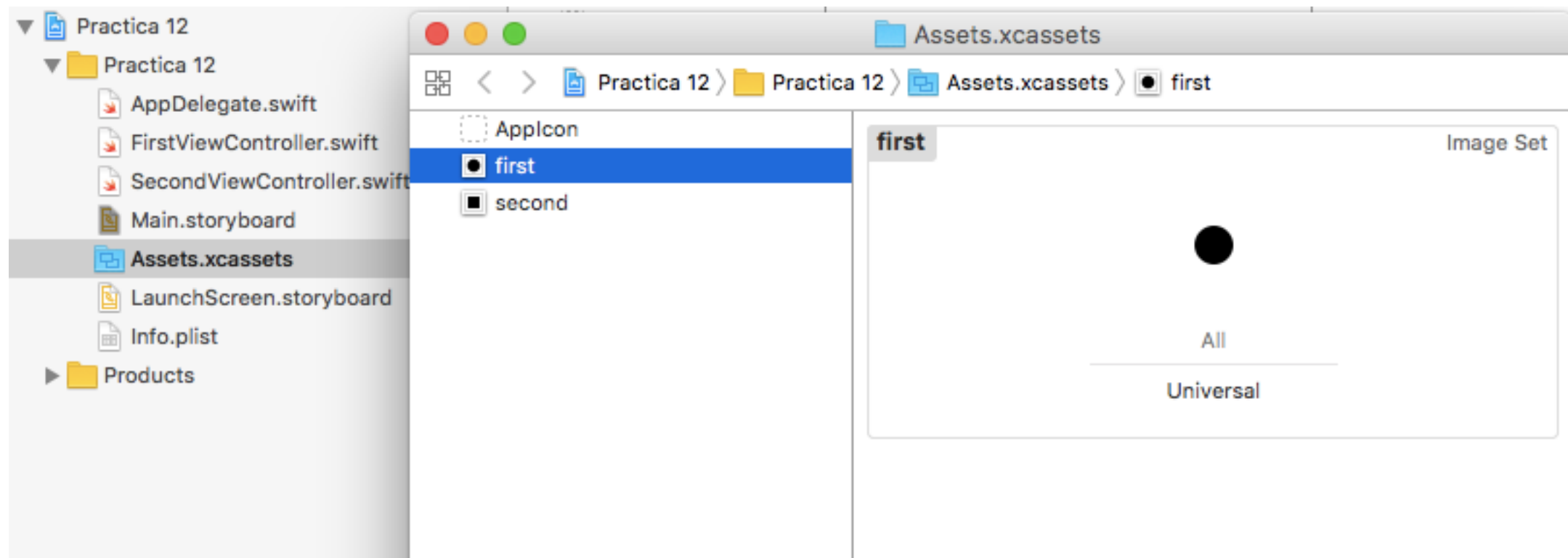
Loaded by FirstViewController



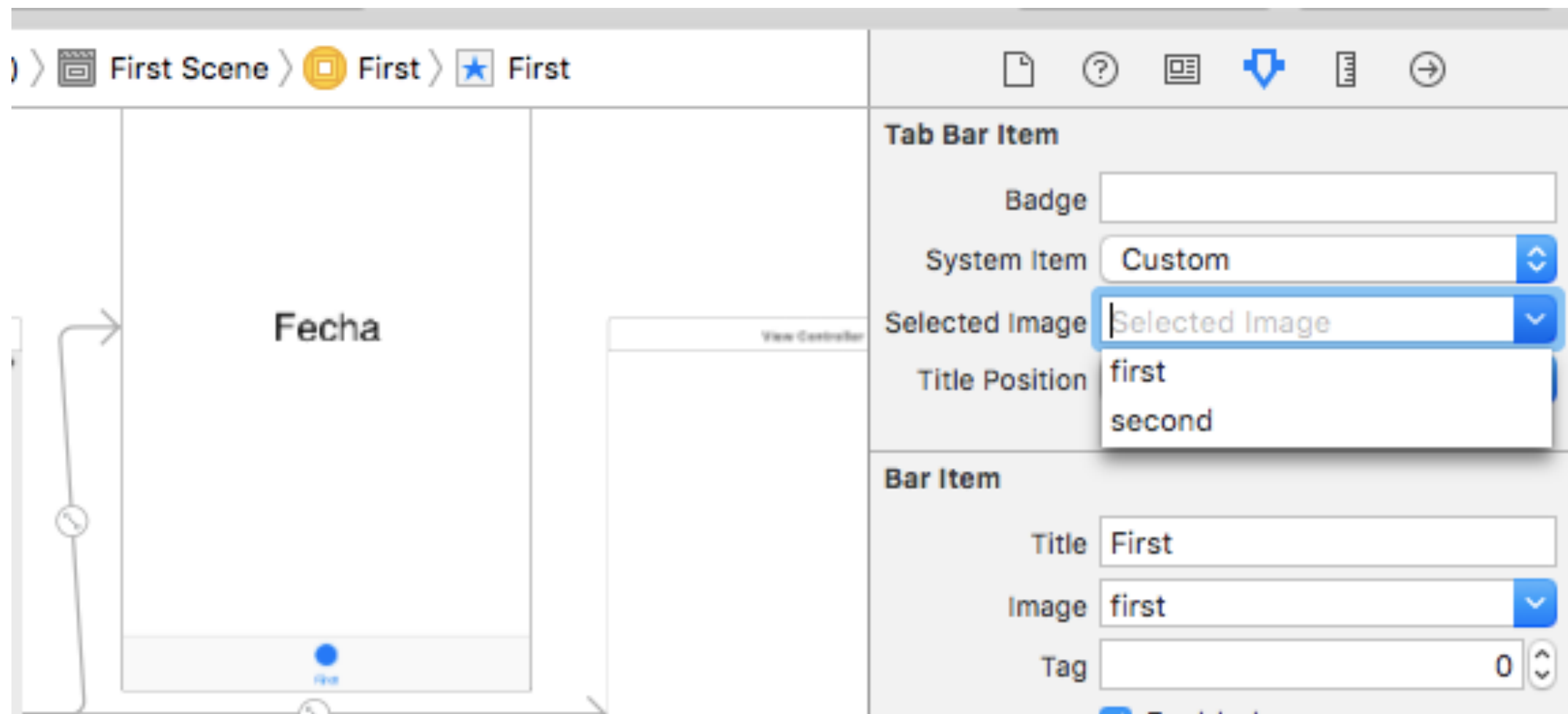
AGREGANDO ICONOS

.....

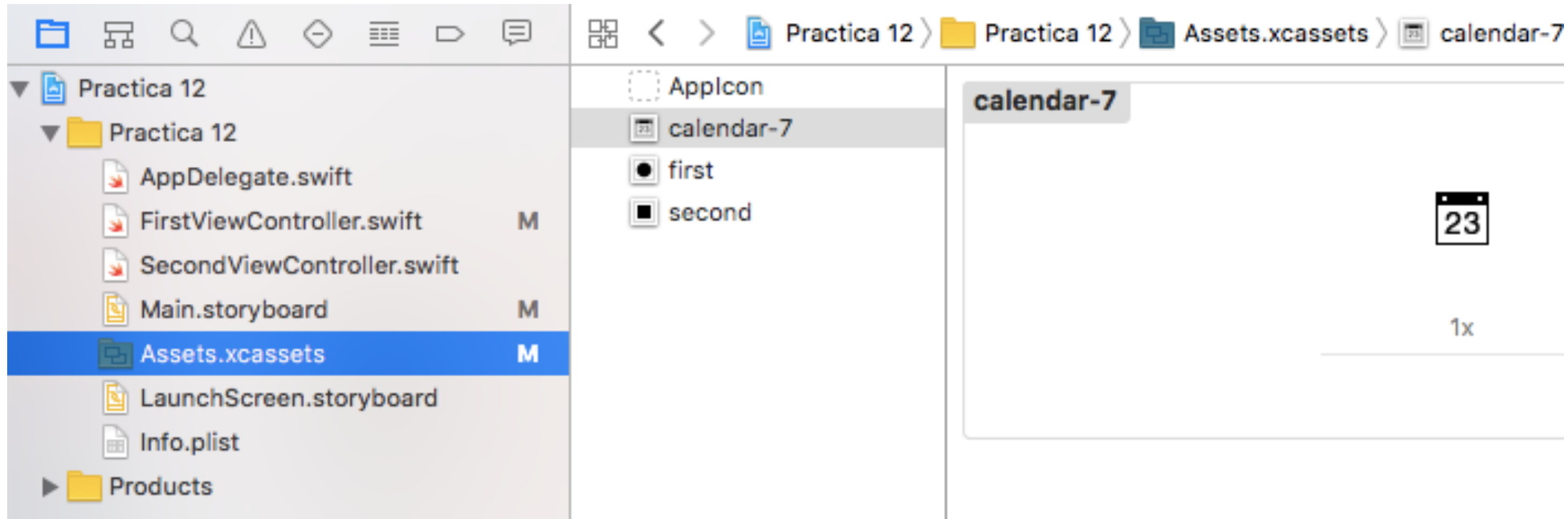
- Los iconos personalizados de la aplicación se encuentran en la carpeta Assets.xcassets.



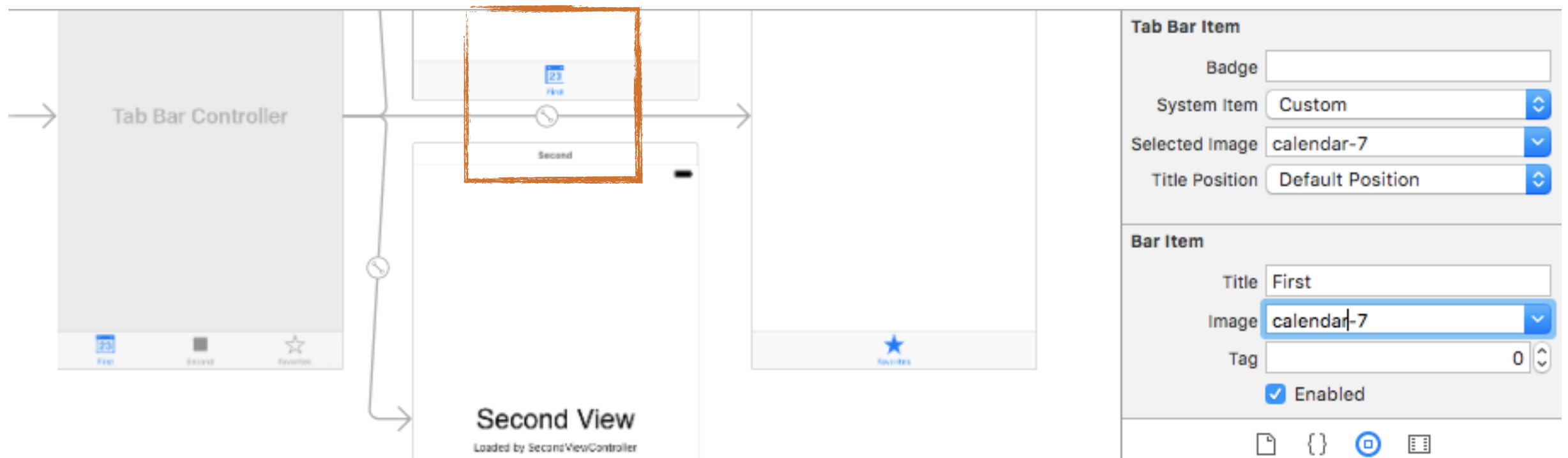
- Una vez agregados a esta carpeta, puedes seleccionarlos dando clic en el tab bar, Tab Bar Item en la opción Selected Image.



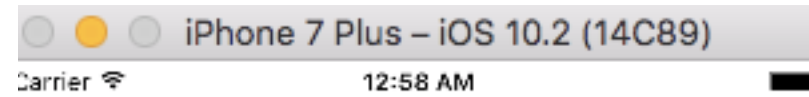
- Busca un nuevo icono para la primera vista, agrégalo a la carpeta Assets



- Cambia el icono de la primera vista.

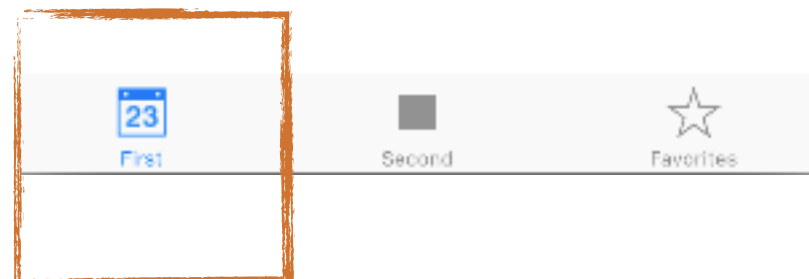


► Ejecuta tu aplicación:



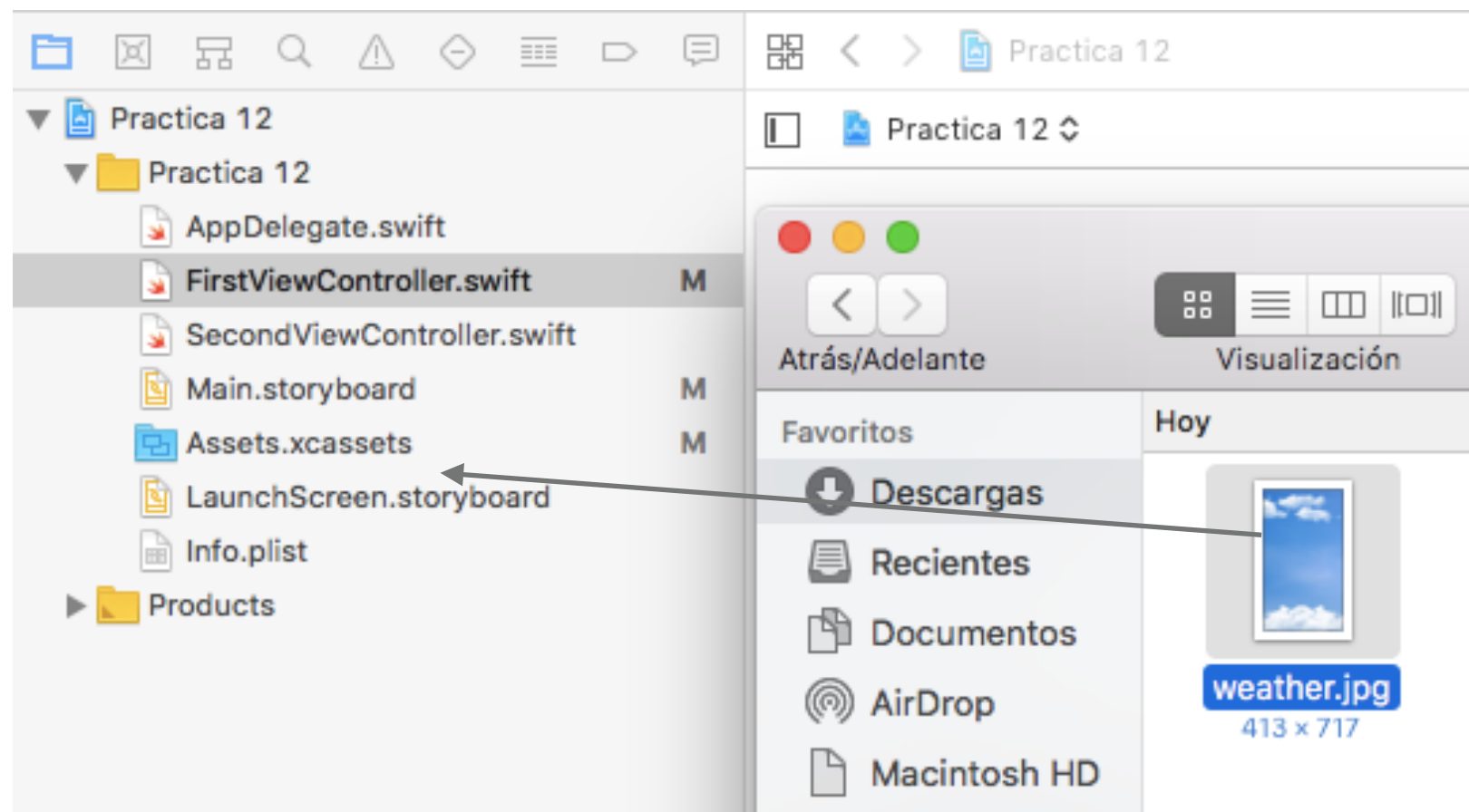
Fecha

Wednesday, March 15, 2017

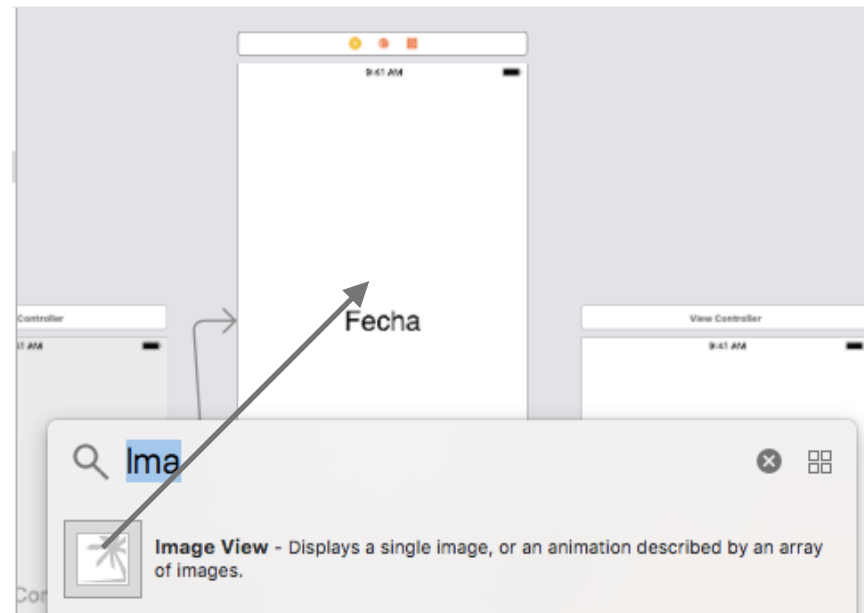


AGREGANDO UN FONDO DE PANTALLA

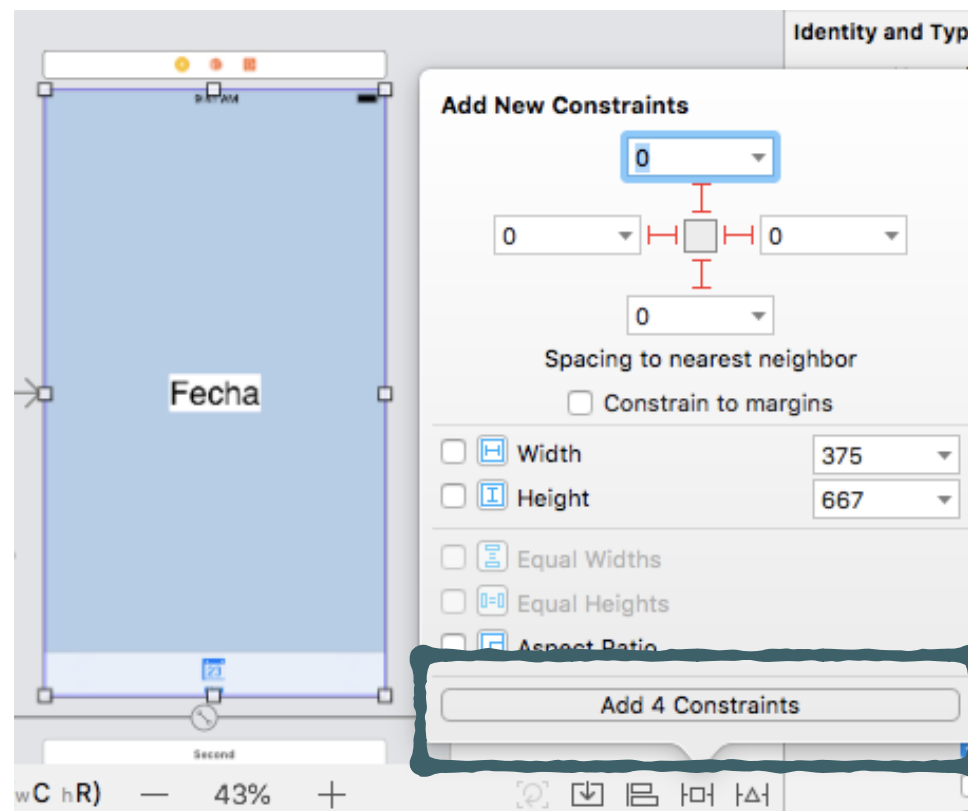
- Busca una imagen para que aparezca como fondo de pantalla de la vista.
- Cópiala a la carpeta de tu práctica.



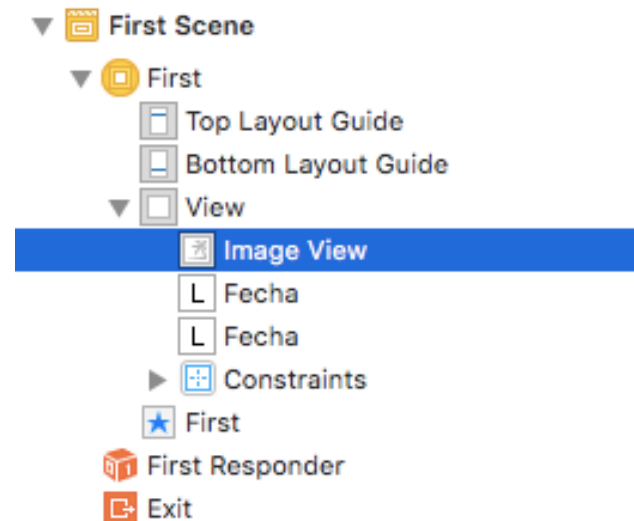
- Agrega a la vista 1, un componente Image View.



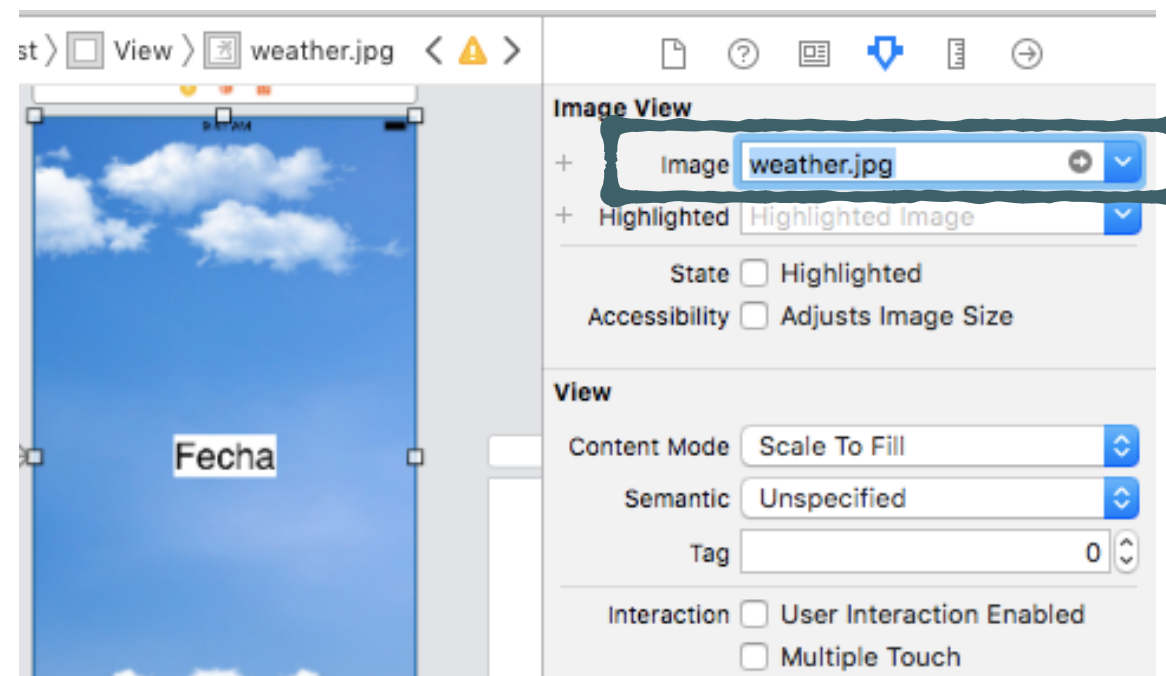
- Agrega las siguientes restricciones:



- Verifica que la jerarquía del componente UIImageView esté por encima de las etiquetas de Fecha.



- Selecciona el componente UIImageView y asigna la imagen correspondiente:



MOSTRANDO LA FECHA

-
- El archivo `FirstViewController.swift` tiene el control de la primera vista, agrega la siguiente función, al ejecutar tu aplicación la salida será mostrada en la consola y en la etiqueta:

```
import UIKit

class FirstViewController: UIViewController {

    @IBOutlet weak var fecha: UILabel!

    override func viewDidLoad() {
        super.viewDidLoad()
        obtenerFecha()
        // Do any additional setup after loading the view, typically from a nib.
    }

    func obtenerFecha(){
        let date = Date()
        let dateFormatter = DateFormatter()

        dateFormatter.dateStyle = .full

        print(dateFormatter.string(from: date))
        fecha.text = dateFormatter.string(from: date)
    }

    override func didReceiveMemoryWarning() {
        super.didReceiveMemoryWarning()
    }
}
```



Wednesday, September 11, 2019
Message from debugger: Terminated due to
signal 9

- Tu aplicación lucirá así:



PRÁCTICA

- La primera vista de la aplicación mostrará la **fecha y hora** del sistema.
- La segunda vista mostrará una **alarma** que le permita al usuario seleccionar la hora deseada. Una vez que la hora coincida con la del sistema, se lanzará un sonido y una alerta con la opción de aceptar que finalizará la reproducción del sonido.
- Finalmente, la tercera vista será un temporizador, (horas/minutos/segundos). Con las opciones de Iniciar, Pausar y Reiniciar.

- Además la aplicación debe contener:
 - Launch Screen.
 - Un fondo diferente para cada vista.
 - Y las restricciones necesarias para cada uno de los componentes.