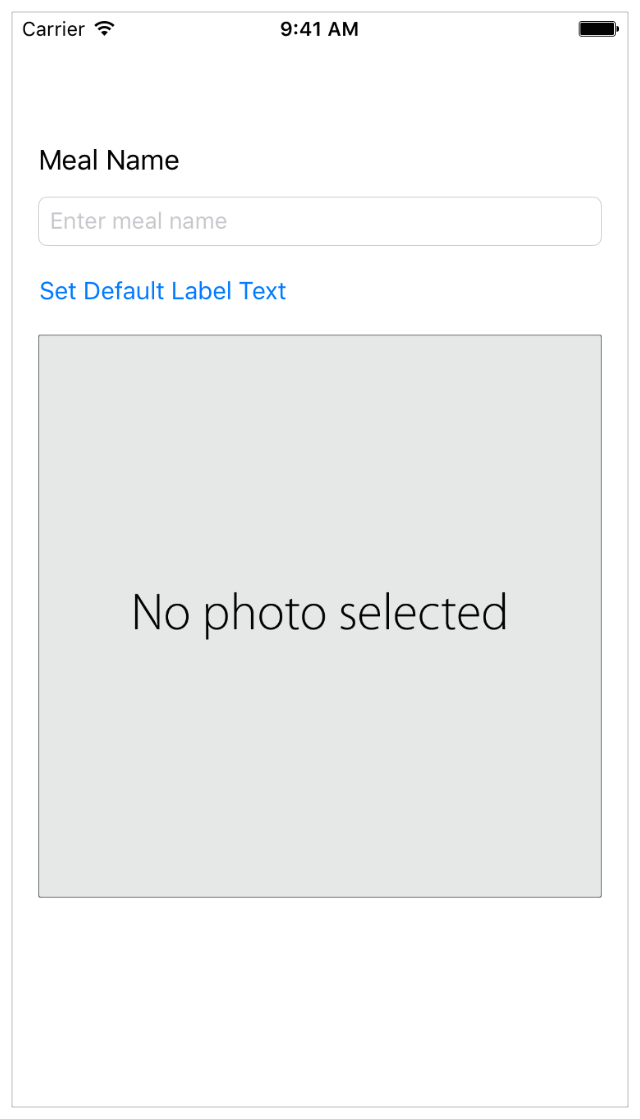
## **Trabajando con View Controllers**



### 

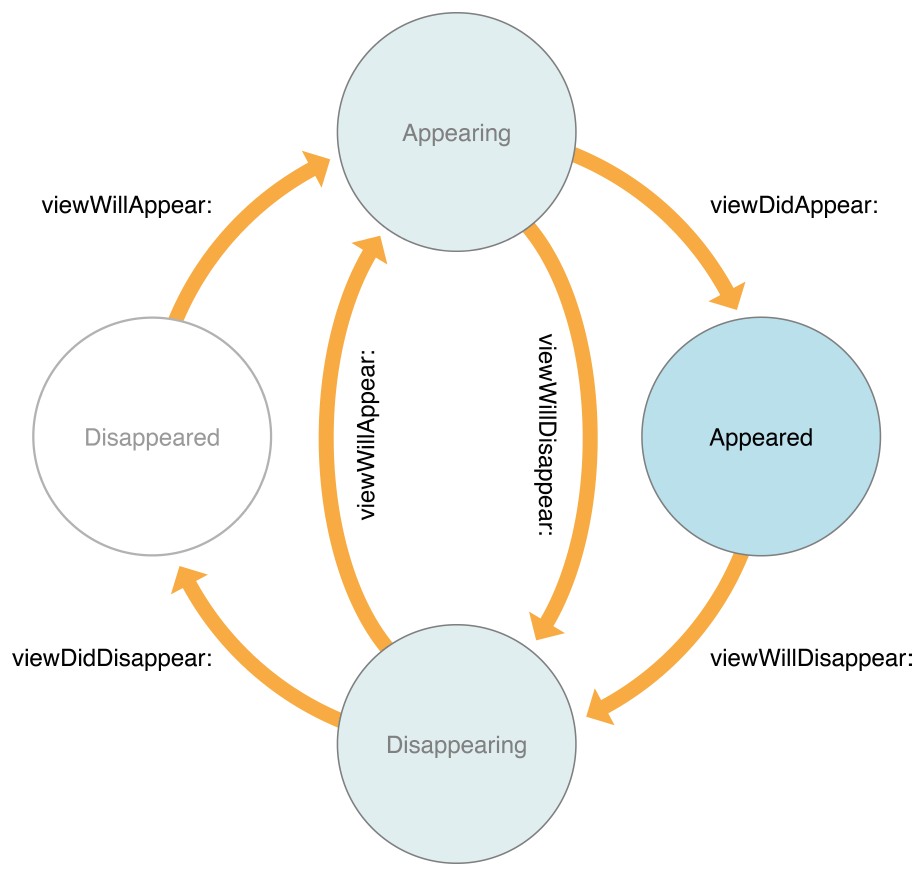
### **Objectivos**

En esta lección aprenderemos:

* El ciclo de vida de un Controlador, enteder los metodos viewDidLoad, viewWillAppear y viewDidAppear
* Pasar Datos entre Controladores
* Salir de un Controlador
* Utilizar gesture events
* Uso de asset Catalog para poner recursos de imagenes

### **View Controller Lifecycle**

Un Controlador tiene el siguiente ciclo de vida:



UIViewController los métodos que se llaman son los siguientes:

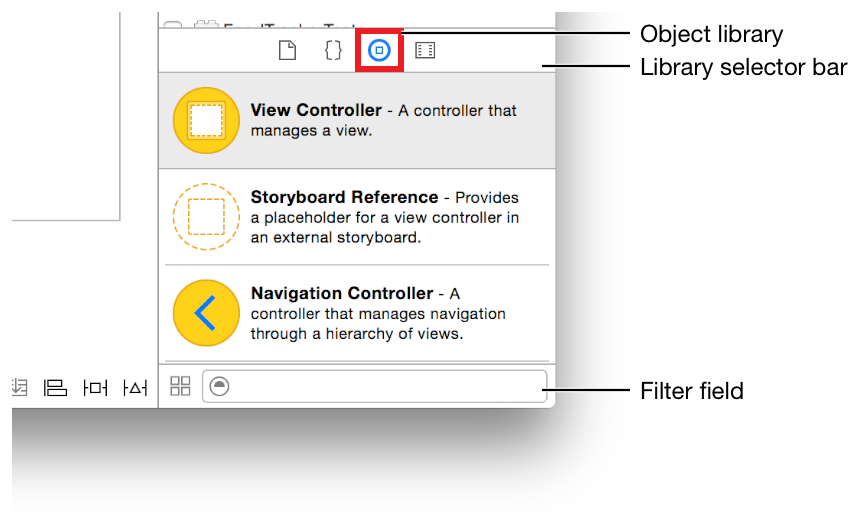
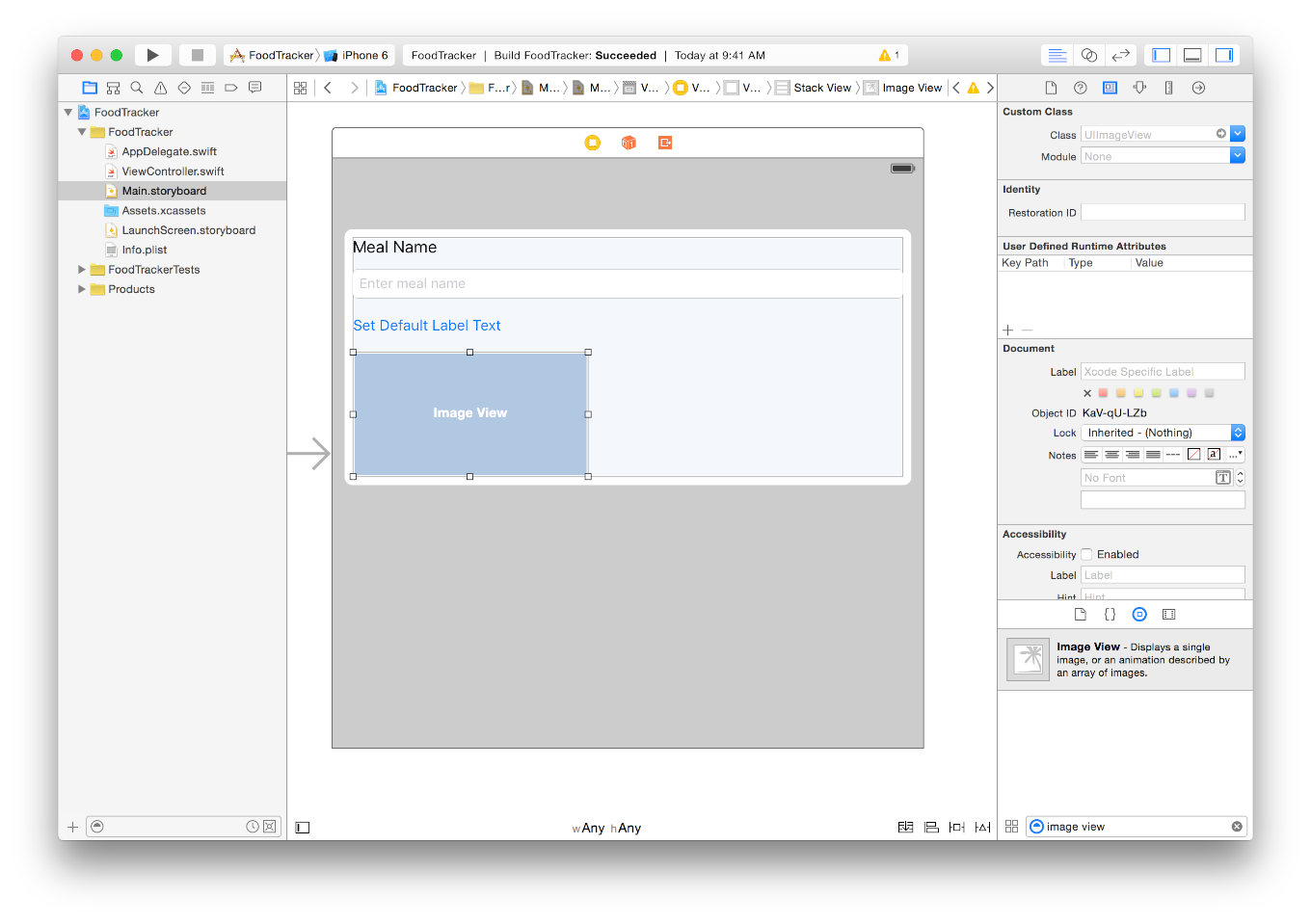
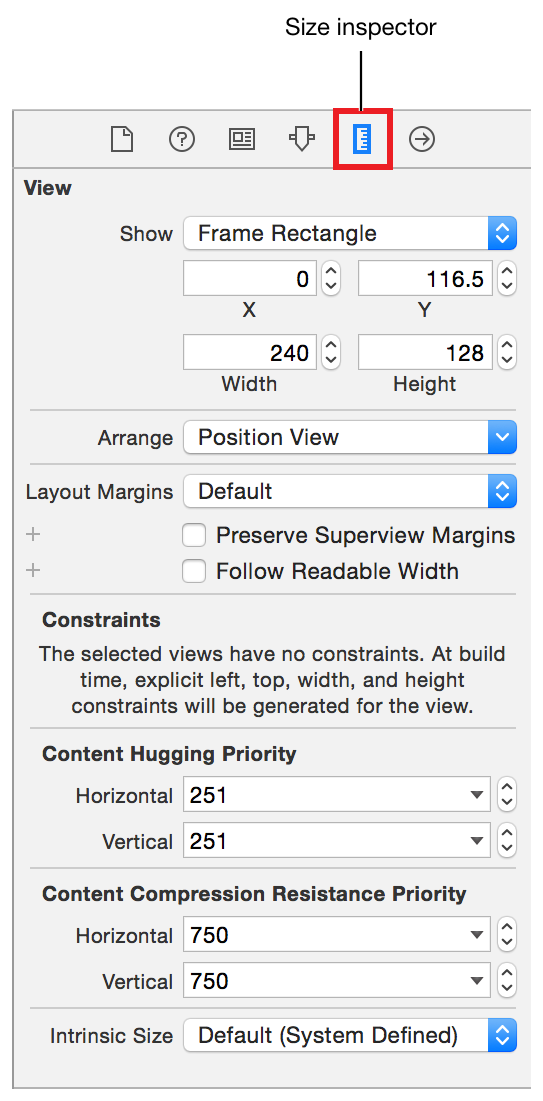
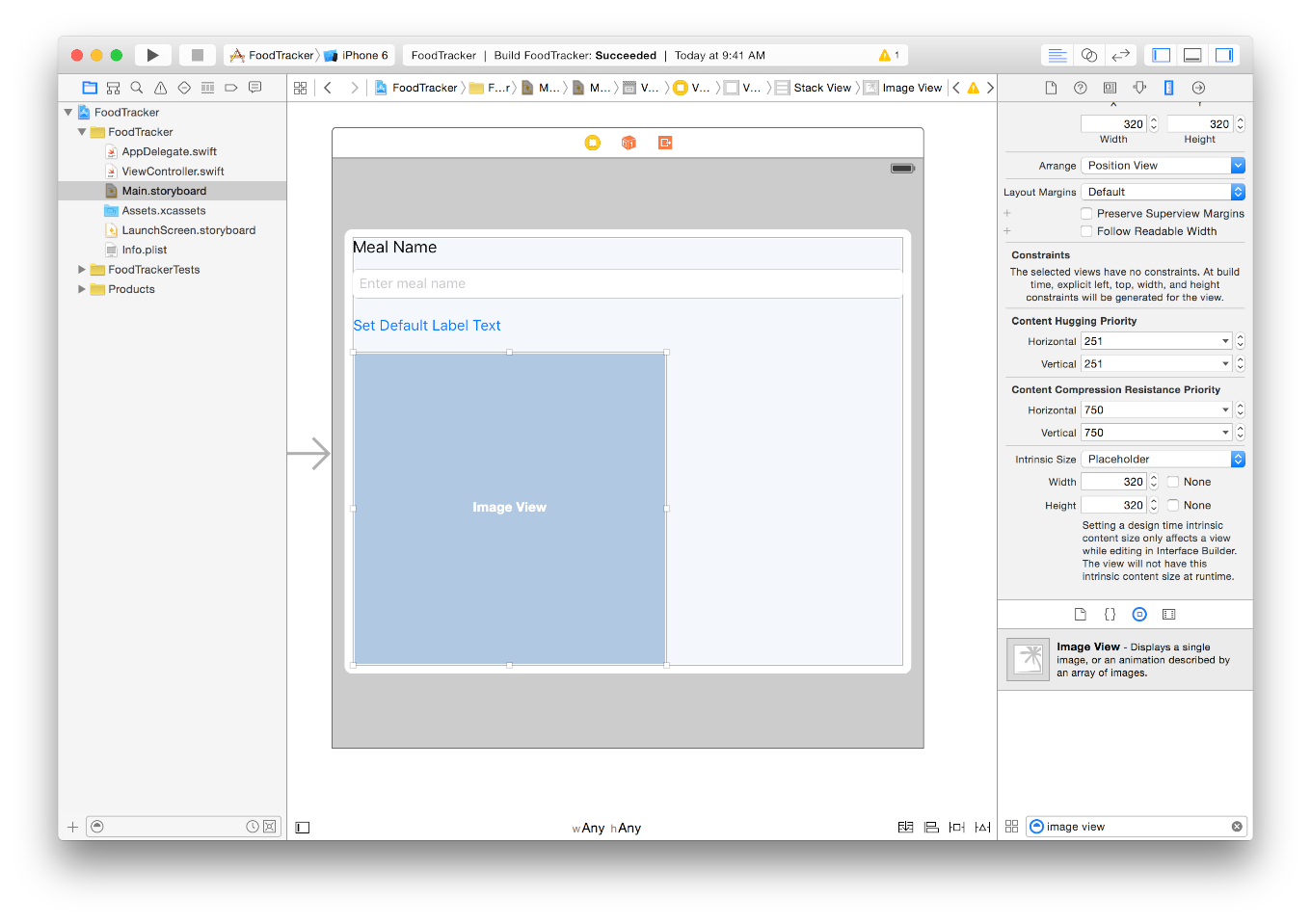
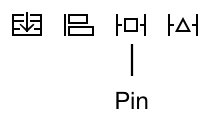
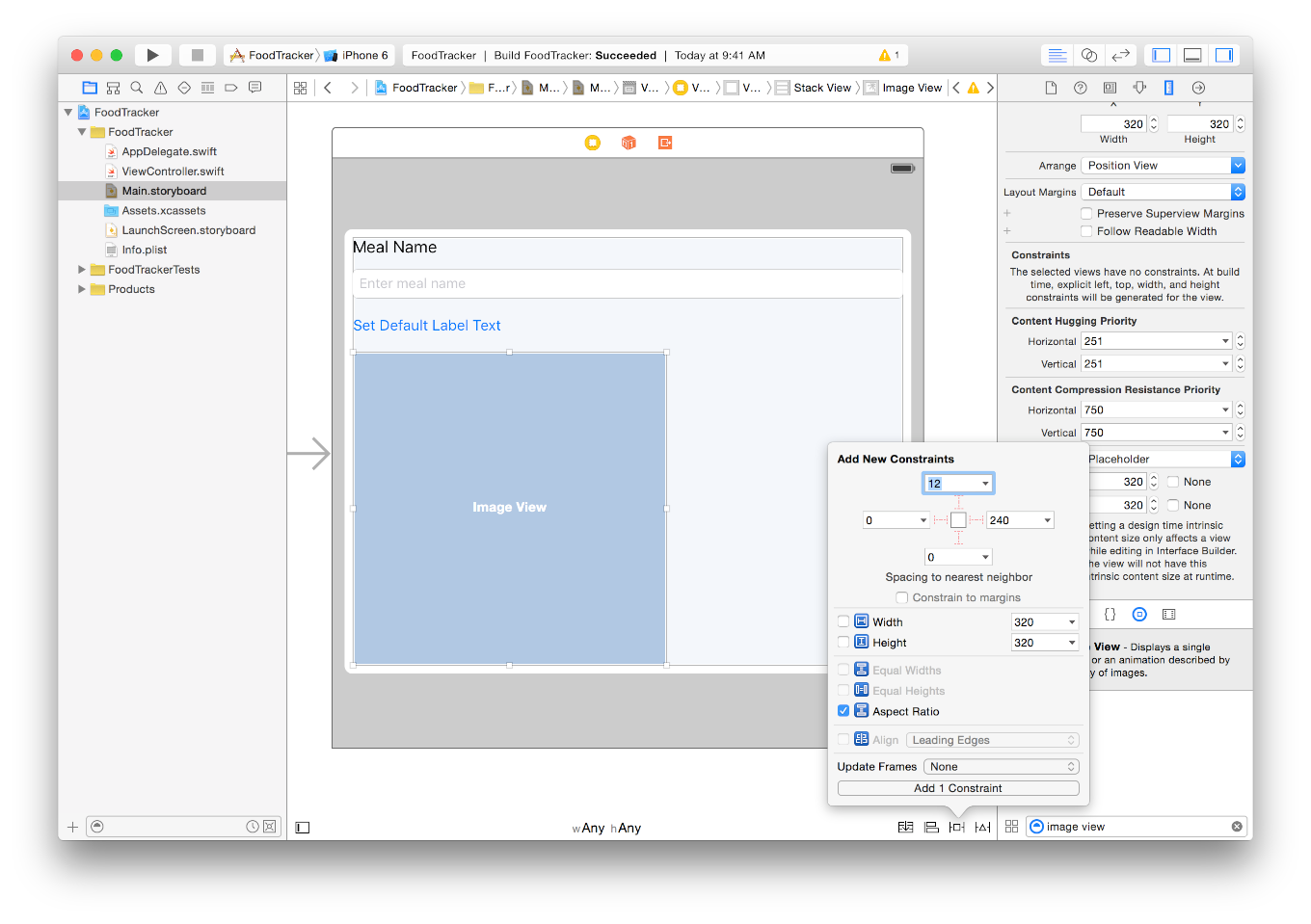
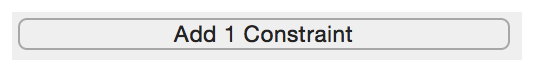
* viewDidLoad()—Este Metodo se llama cuando se termina cargar o crear todos los elementos del storyboard.
* viewWillAppear()—Este Método se llama cuando la vista está lista y esta apunto de aparecer, y esta antes del método viewDidLoad()
* viewDidAppear()—Destinados a cualquiera de las operaciones que desea que se produzca tan pronto como la vista se hace visible, como ir a buscar los datos o mostrar una animación. Debido a la visibilidad de un punto de vista puede ser activado o oscurecida por otros puntos de vista, este método siempre se llama inmediatamente después de la vista de contenido aparece en la pantalla.

Usted puede utilizar estos metodos en el momento adecuado por ejemplo cuando se quiere delegar los eventos Input Text del TextField component:

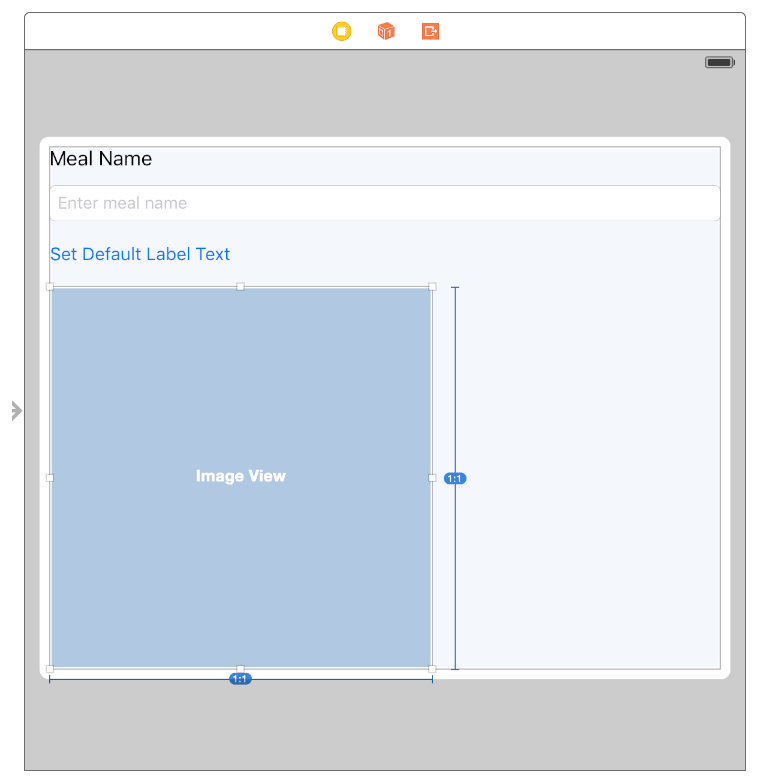
* override func viewDidLoad() {
* super.viewDidLoad()
* // Handle the text field’s user input through delegate callbacks.
* nameTextField.delegate = self
* }

### **Agregar Image container**

**Para añadir una imageview a la escena**

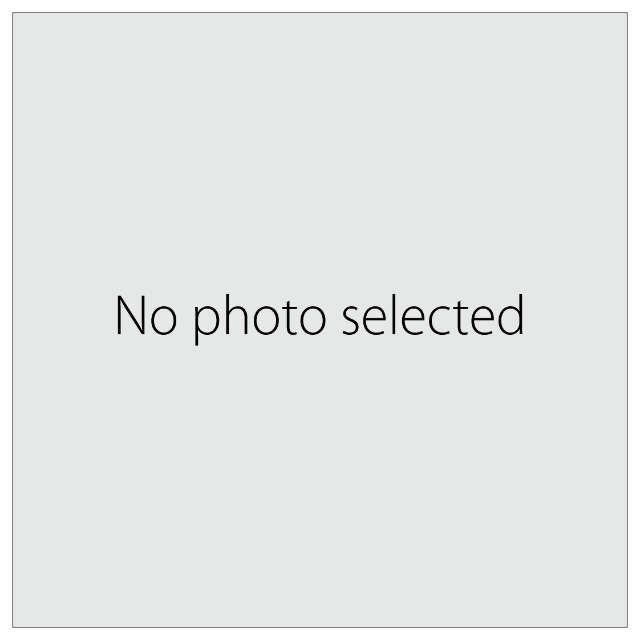
1. Abrir su [storyboard](https://developer.apple.com/library/ios/referencelibrary/GettingStarted/DevelopiOSAppsSwift/GlossaryDefinitions.html#//apple_ref/doc/uid/TP40015214-CH12-SW8), Main.storyboard.
2. Abrir el [Object library](https://developer.apple.com/library/ios/referencelibrary/GettingStarted/DevelopiOSAppsSwift/GlossaryDefinitions.html#//apple_ref/doc/uid/TP40015214-CH12-SW54) en el [utility area](https://developer.apple.com/library/ios/referencelibrary/GettingStarted/DevelopiOSAppsSwift/GlossaryDefinitions.html#//apple_ref/doc/uid/TP40015214-CH12-SW72). (Alternativa: View > Utilities > Show Object Library.)
3. 
4. En el Object Library buscar image view Ui component
5. Arrastre el Image View object del Object library as su scena debajo del Boton tal como se muestra en la siguiente figura:
6. 
7. Con el Image View seleccionando abra el “Size Inspector”.
8. [Size inspector](https://developer.apple.com/library/ios/referencelibrary/GettingStarted/DevelopiOSAppsSwift/GlossaryDefinitions.html#//apple_ref/doc/uid/TP40015214-CH12-SW82) se encarga de modificar las dimensiones en su storyboard.
9. 
10. rellene con 320 los campos Width y Height de la seccion [intrinsic content size](https://developer.apple.com/library/ios/referencelibrary/GettingStarted/DevelopiOSAppsSwift/GlossaryDefinitions.html#//apple_ref/doc/uid/TP40015214-CH12-SW110). presione Enter.
11. 
12. Abrir Pin menu que se encuentra en la parte inferior del Story board editor.
13. 
14. seleccionar el checkbox **Aspect Ratio**.
15. El Pin menu debería mostrarse como lo siguiente:
16. 
17. en el Pin menu, click en Add 1 Constraints button.
18. 
19. Ahora su Imageview tiene la escala 1:1 aspect ratio,lo cual significa que siempre será un cuadrado.

Su UI debería mostrarse de la siguiente manera :



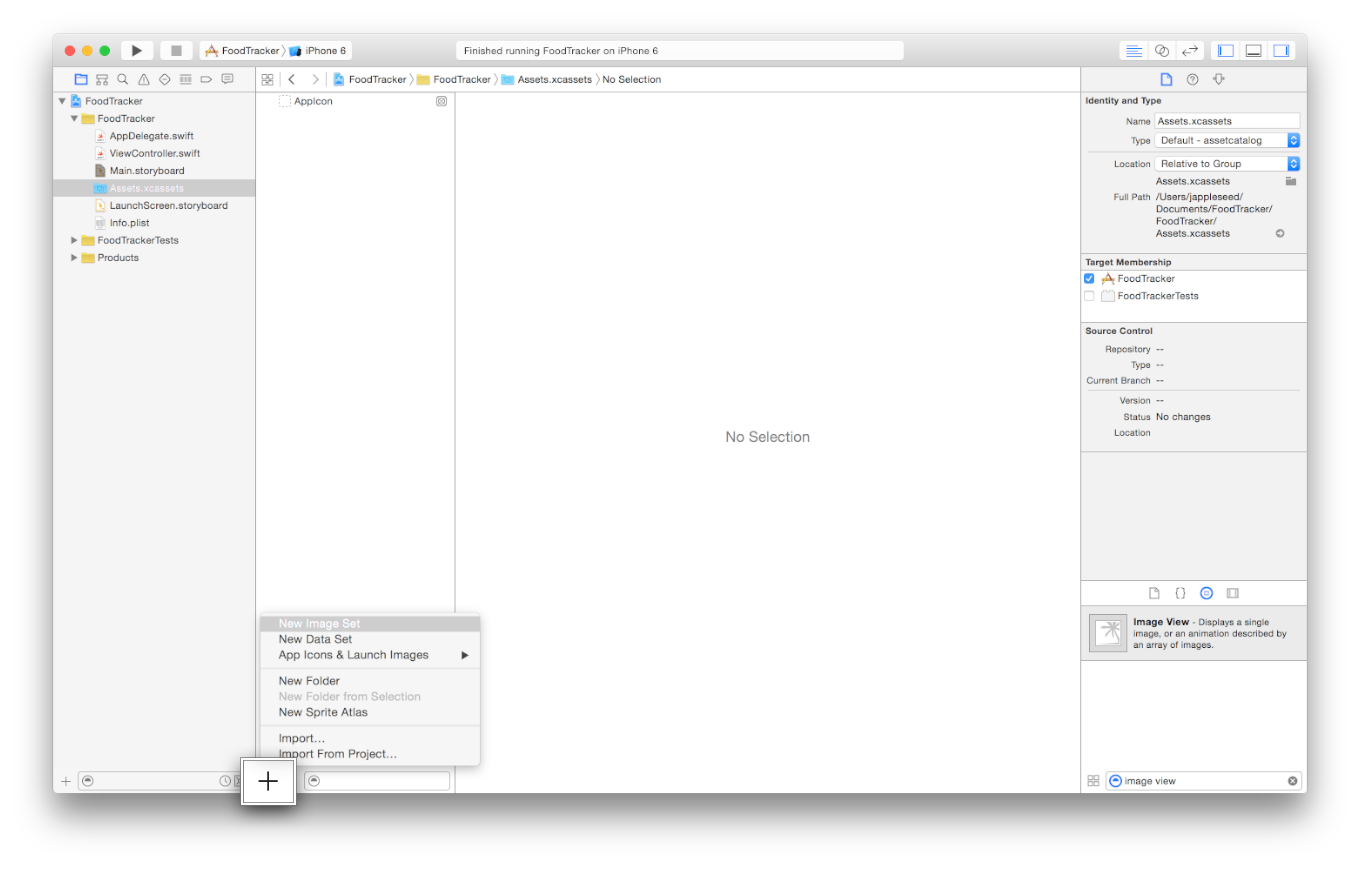
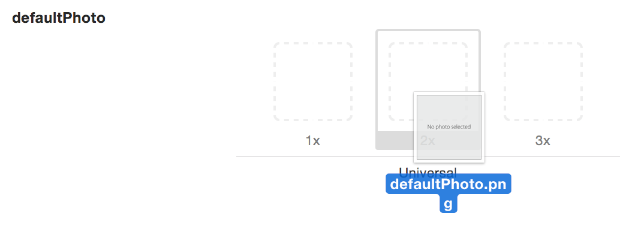
### **Mostrar una Foto por defecto**

Los usuarios necesitan una indicación de que pueden interactuar con la vista de la imagen para seleccionar una foto. Para lograr esto, añadir un marcador de posición de imagen predeterminada que se comunica a los usuarios que pueden elegir una foto.



Puede encontrar la imagen mostrada anteriormente en el archivo / carpeta del archivo descargable al final de esta lección, o utilizar su propia imagen.

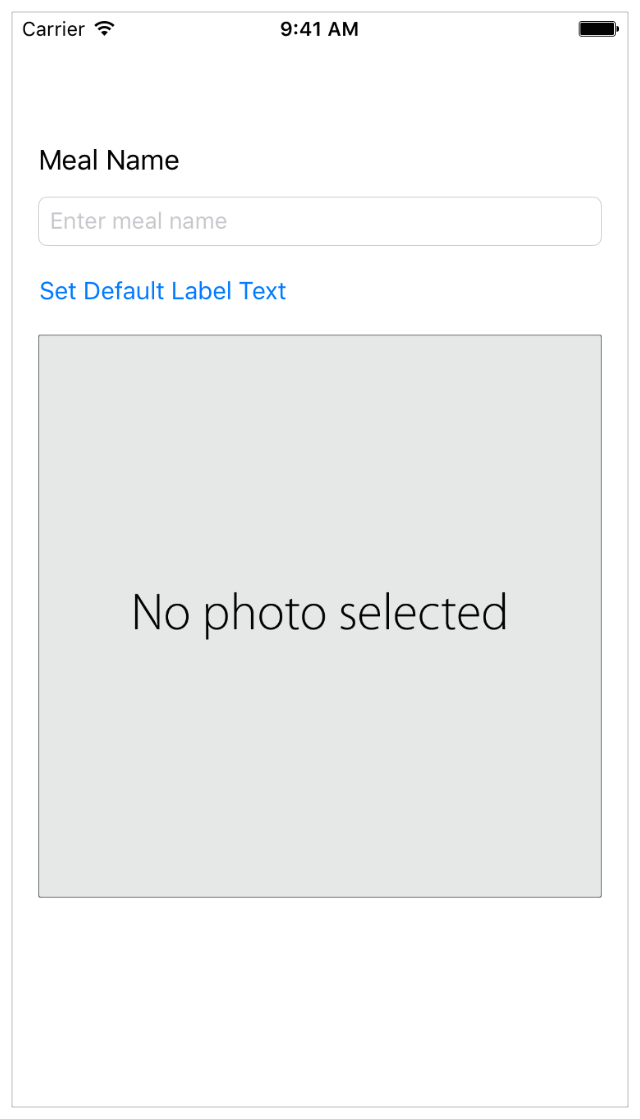
**Para añadir una imagen a su proyecto**

1. dentro de [project navigator](https://developer.apple.com/library/ios/referencelibrary/GettingStarted/DevelopiOSAppsSwift/GlossaryDefinitions.html#//apple_ref/doc/uid/TP40015214-CH12-SW57), seleccione Assets.xcassets para ver el catálogo de recursos
2. El Assets.xcassets catalog es el lugar donde se guarda los recursos ya sean imágenes por ejemplo.
3. En la parte inferior seleccione el boton (+) el cual abrira un popup para seleccionar el tipo de recurso.
4. 
5. Doble click en la imagen selecciona y renombrar por defaultPhoto.
6. en su computadora seleccionar la imagen que quiere agregar.
7. Arrastrar hasta el campo 2x y soltar la imagen.
8. 

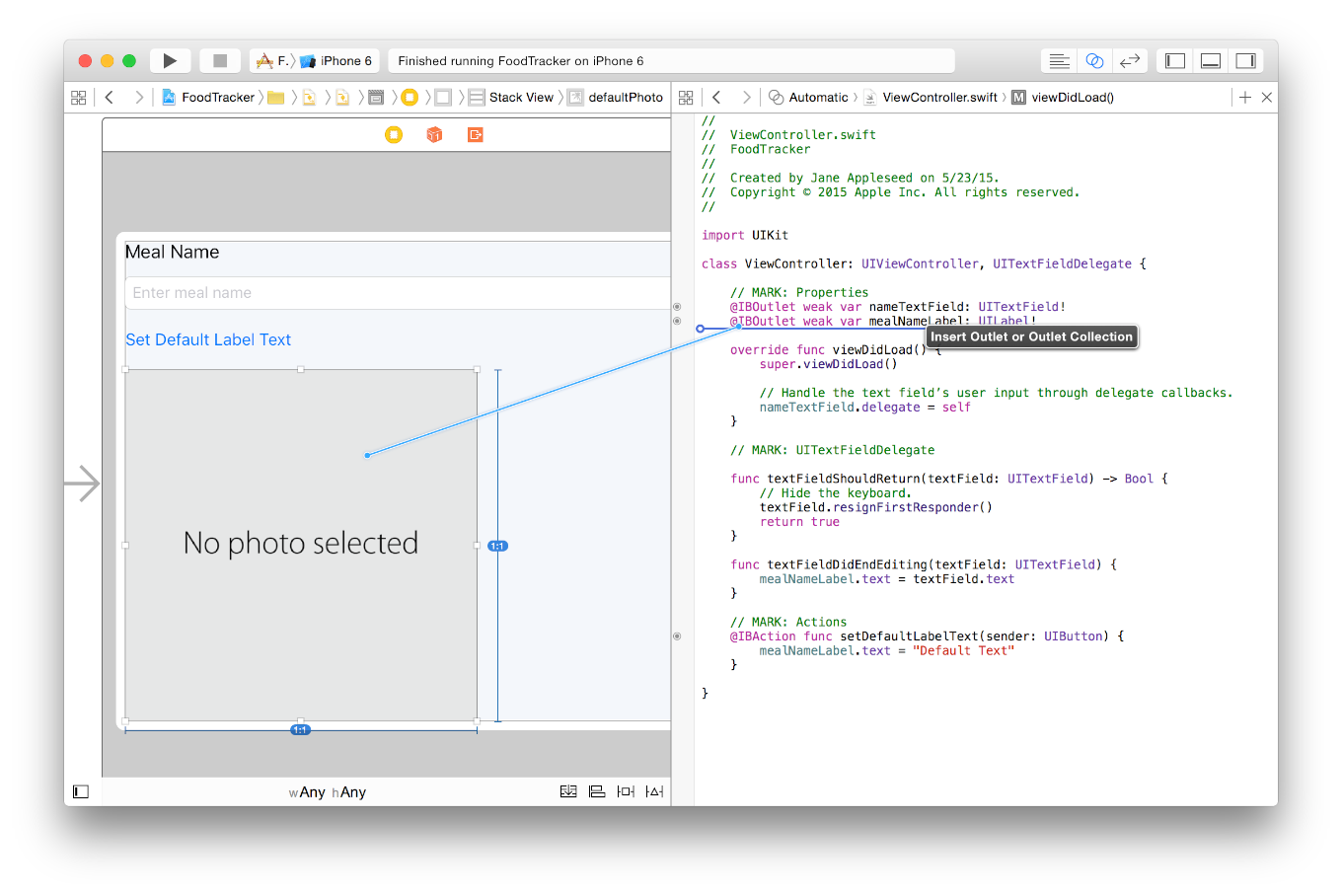
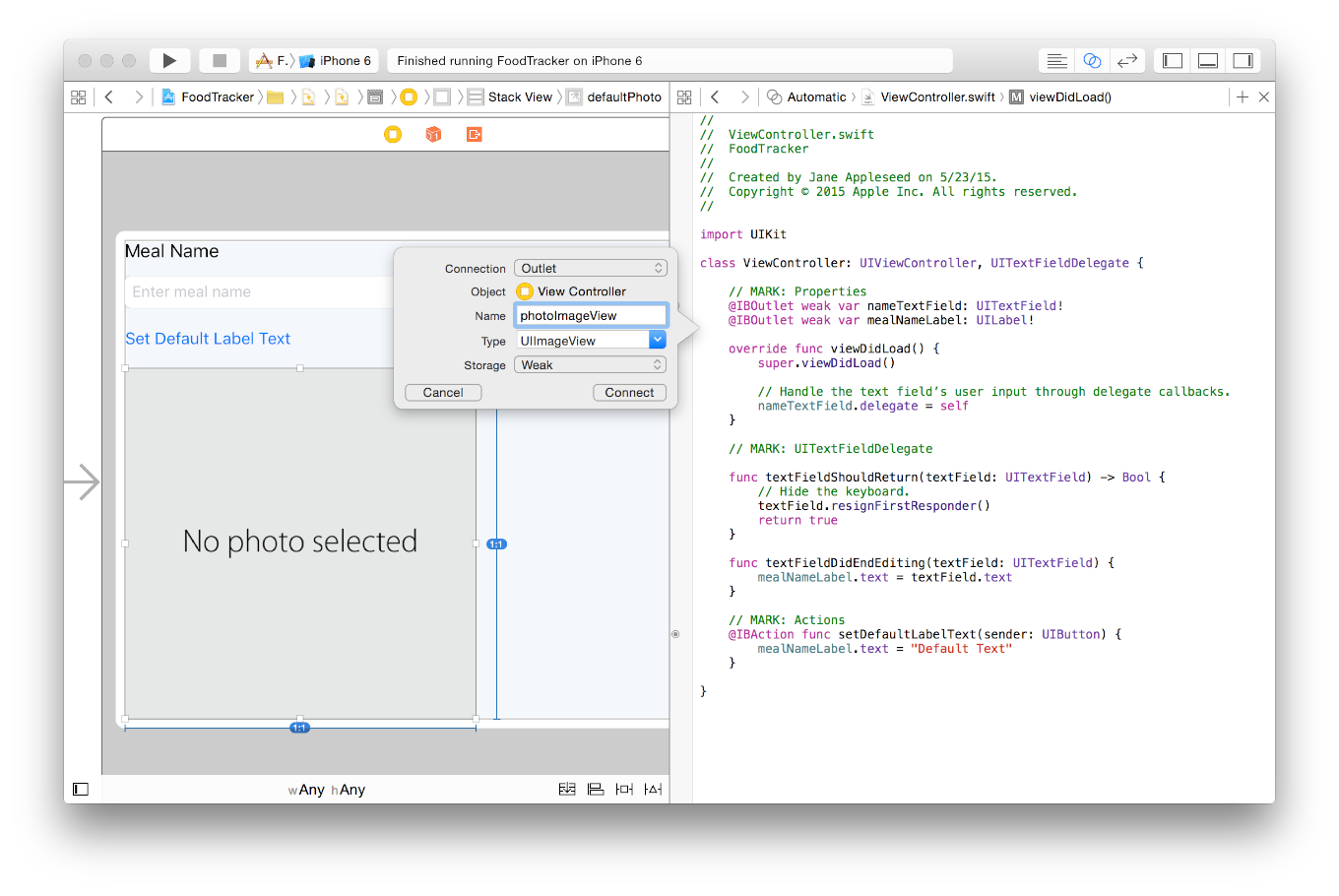
**Para Mostrar la imagen seleccionada en Asset Catalog dentro el Image view**

1. Abrir el Storyboard
2. Seleccionar la imagen view UI component
3. en el attribute inspector en el campo Image seleccionar defaultPhoto.

*Ejecute su aplicación el cual debería mostrarse de la siguiente manera*



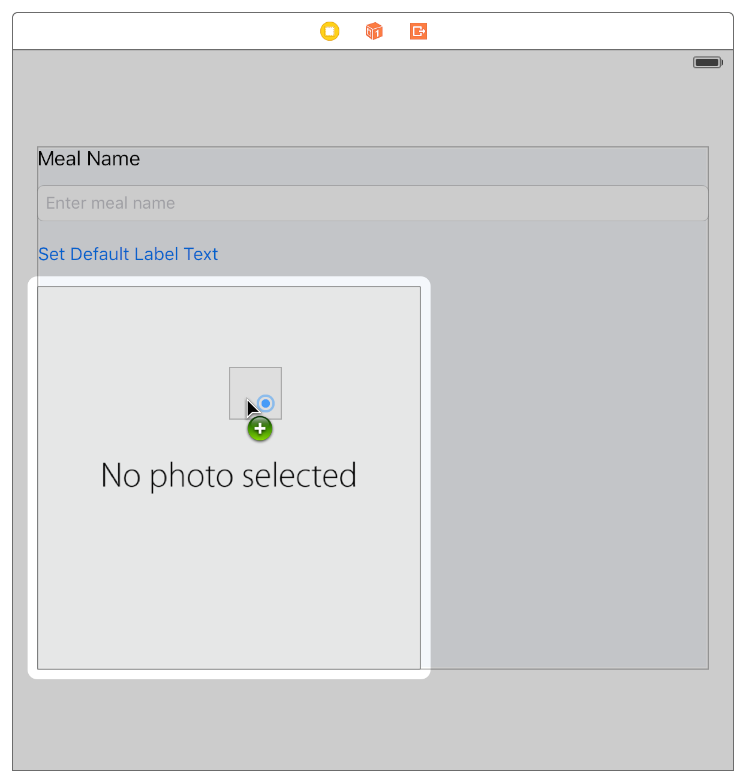
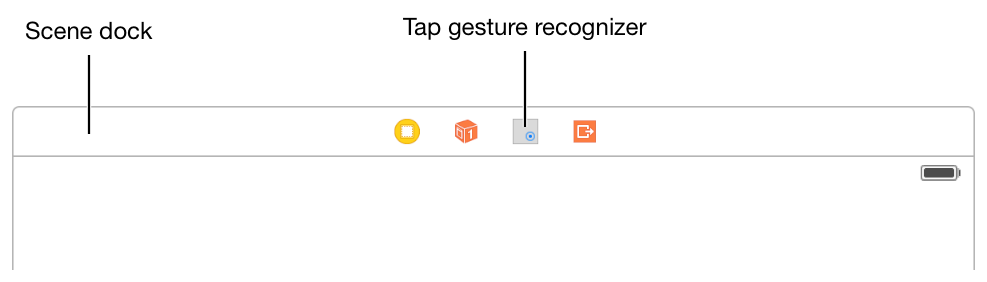
### **Conectar el ImageView con el codigo**

1. Agregar el conector a la clase ViewController.swift.
2. 
3. En el Dialago de connection poner el campo Name a: photoImageView.
4. Su interfaz debería lucir de la siguiente manera:
5. 
6. Click Connect.

### **Crear un Gesture Recognizer**

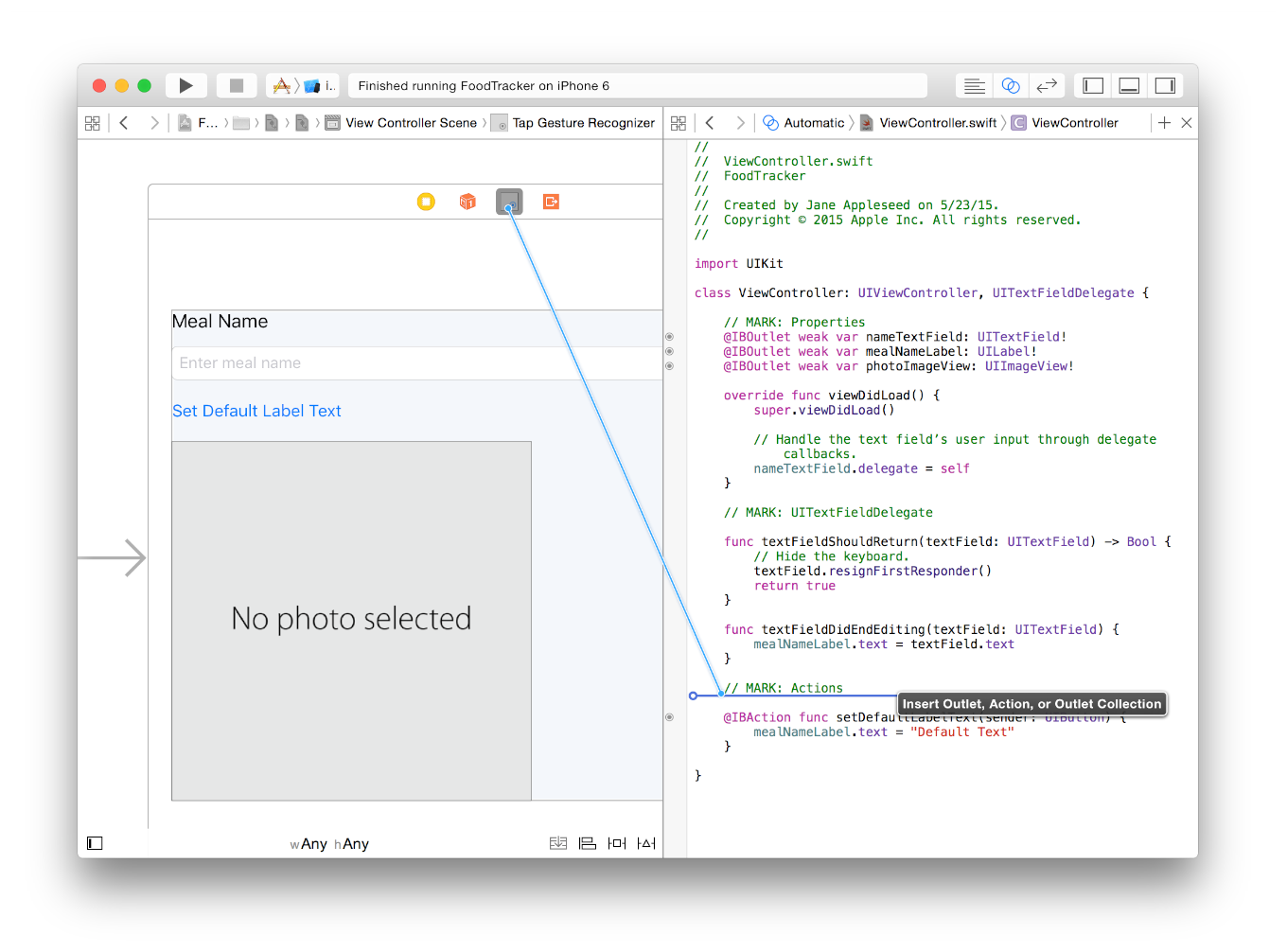
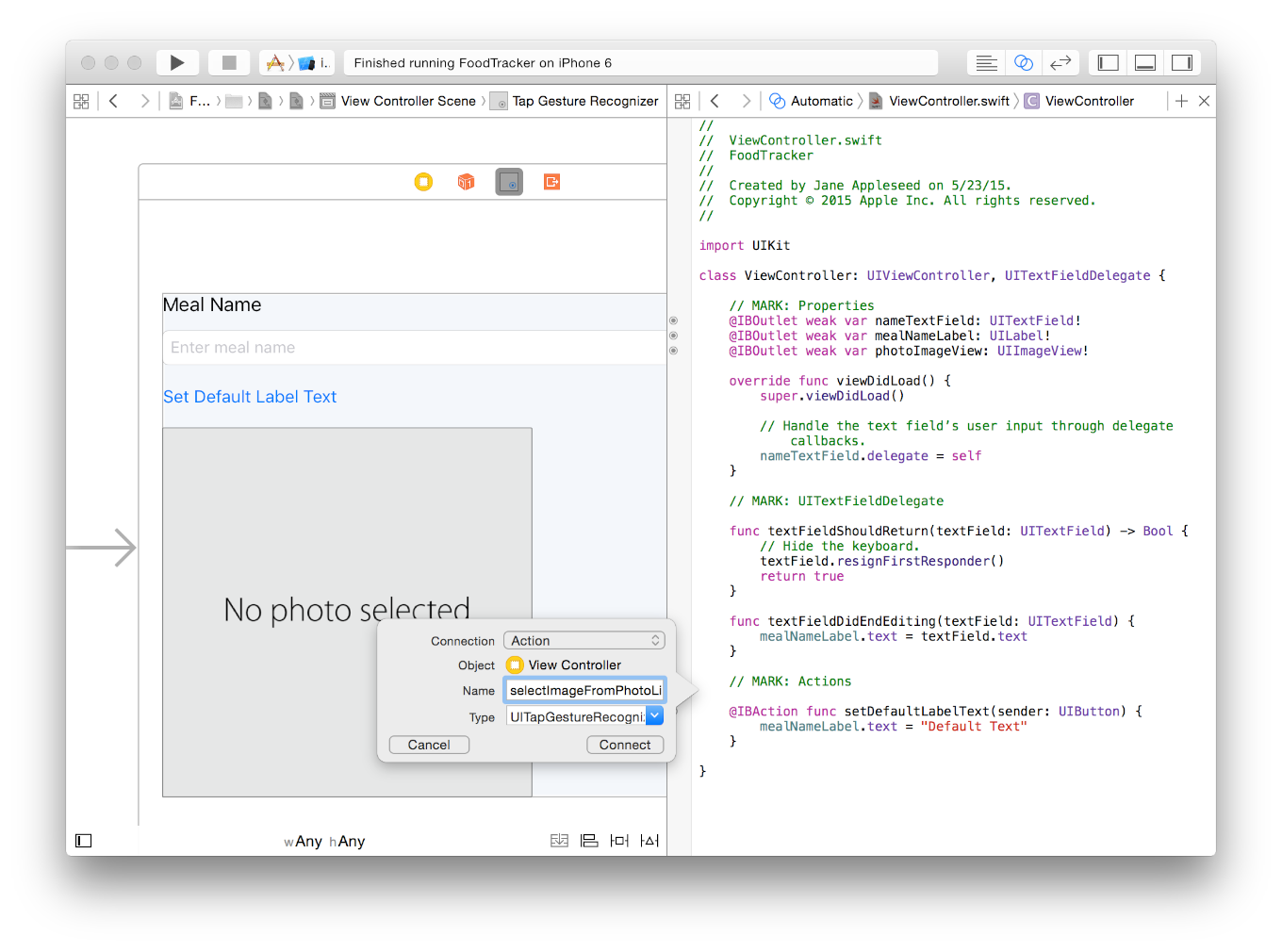
Agregar (UITapGestureRecognizer) al image view, el cual da la funcionalidad de Gesture Reconognizer.

**Para agregar Gesture Recognizer agregue lo siguiente**

1. abrir el Object library. (alternativa: View > Utilities > Show Object Library.)
2. en el Object library, buscar tap gesture
3. Arrastre el gesture del Object Library al content View
4. 
5. el Gesture agregado aparecerá en la [scene dock](https://developer.apple.com/library/ios/referencelibrary/GettingStarted/DevelopiOSAppsSwift/GlossaryDefinitions.html#//apple_ref/doc/uid/TP40015214-CH12-SW63).
6. 

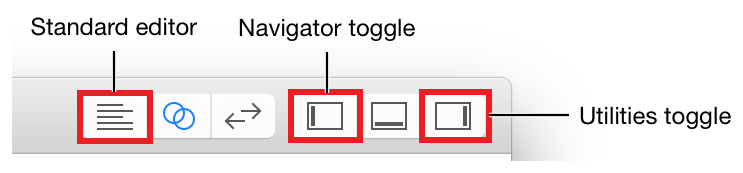
### **Conectar Gesture recognizer al Codigo**

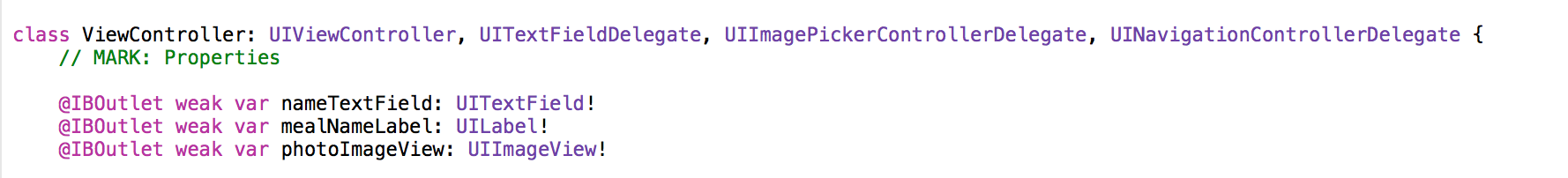
**para conectar gesture recognizer al ViewController.swift code**

1. Arrastre desde el gesture recognizer del Scene dock al codigo en ViewController.swift.
2. 
3. En el diálogo de coneccion llene el campo Nombre: selectImageFromPhotoLibrary.
4. Para can Type seleccione UITapGestureRecognizer.
5. El dialogo debería mostrarse de la siguiente manera:
6. 
7. Click Connect.
8. Xcode agregara al ViewController.swift la siguiente accion
   * @IBAction func selectImageFromPhotoLibrary(sender: UITapGestureRecognizer) {
   * }

### **Crear Image Picker**

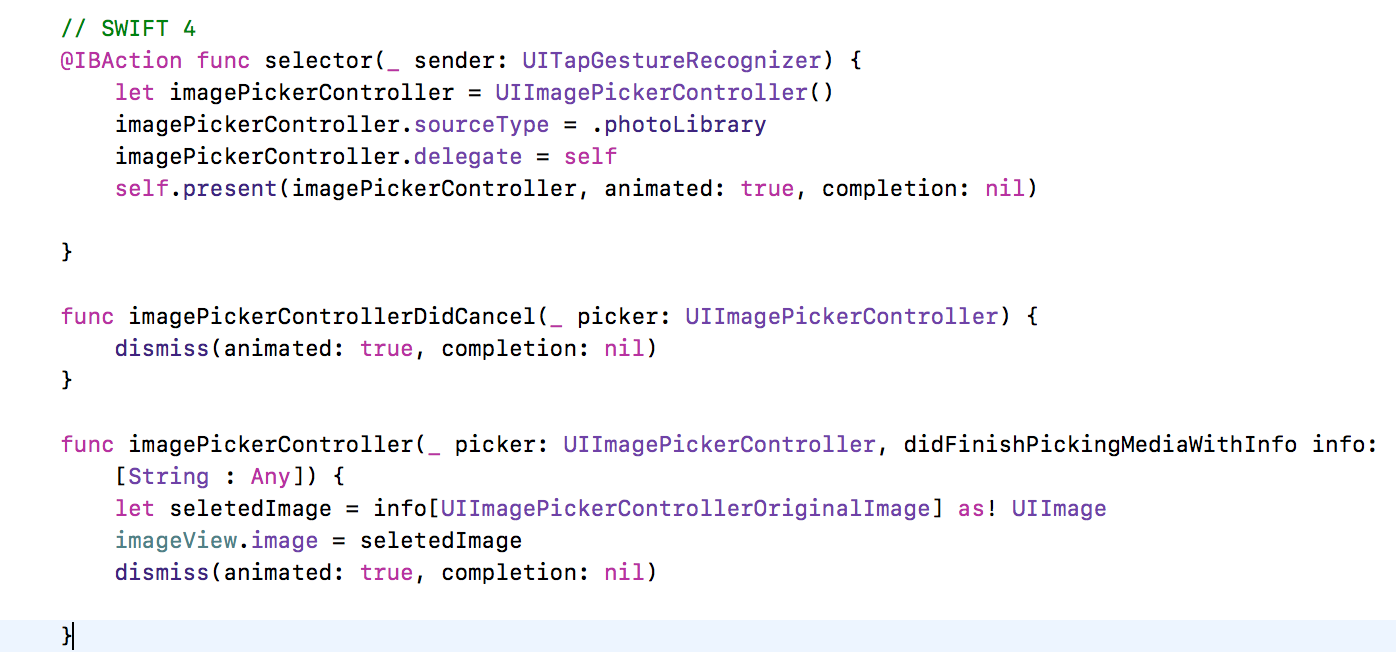
**Adoptar UIImagePickerControllerDelegate y UINavigationControllerDelegate protocols**

1. Volver a configurar Xcode
2. 
3. Agregar he implementar dos delegados : ***UIImagePickerControllerDelegate, UINavigationControllerDelegate***

* 

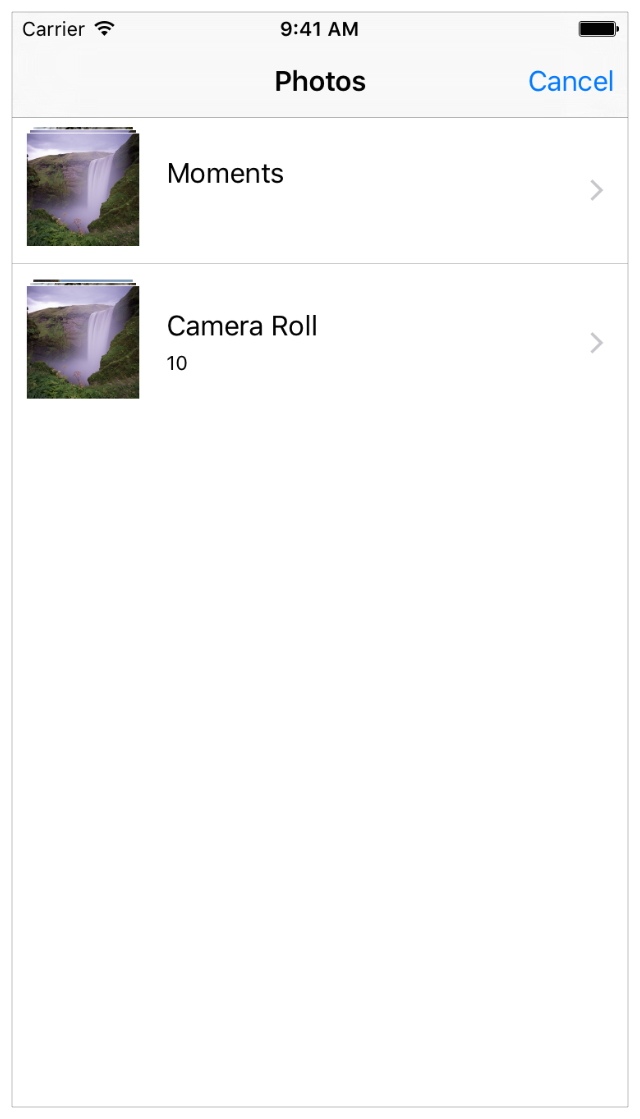
4. Implementar los metodos

//swift 4



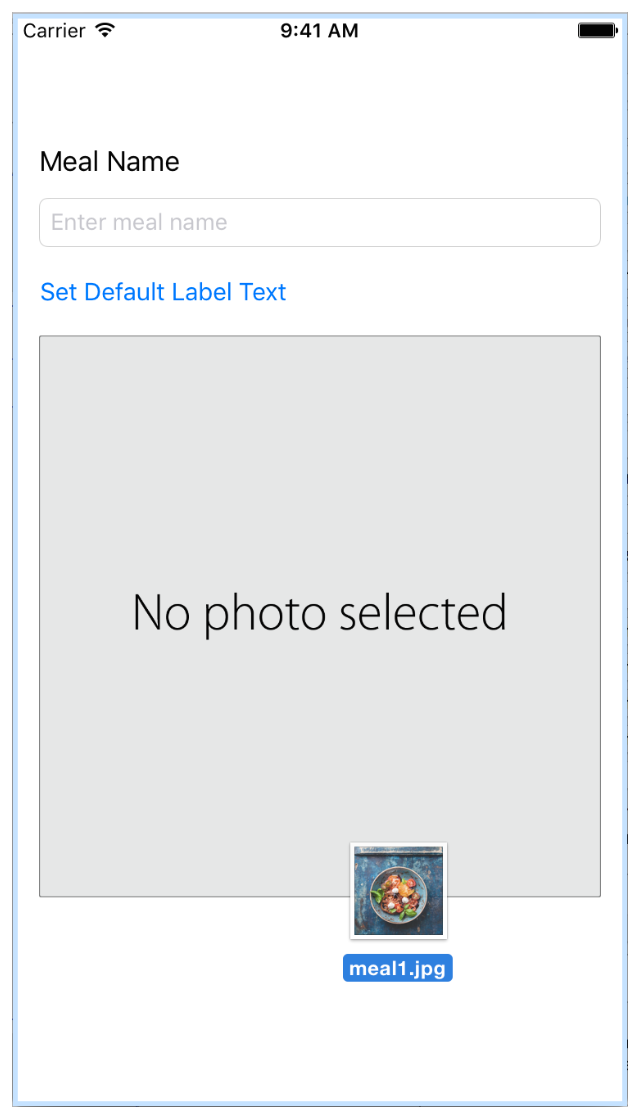
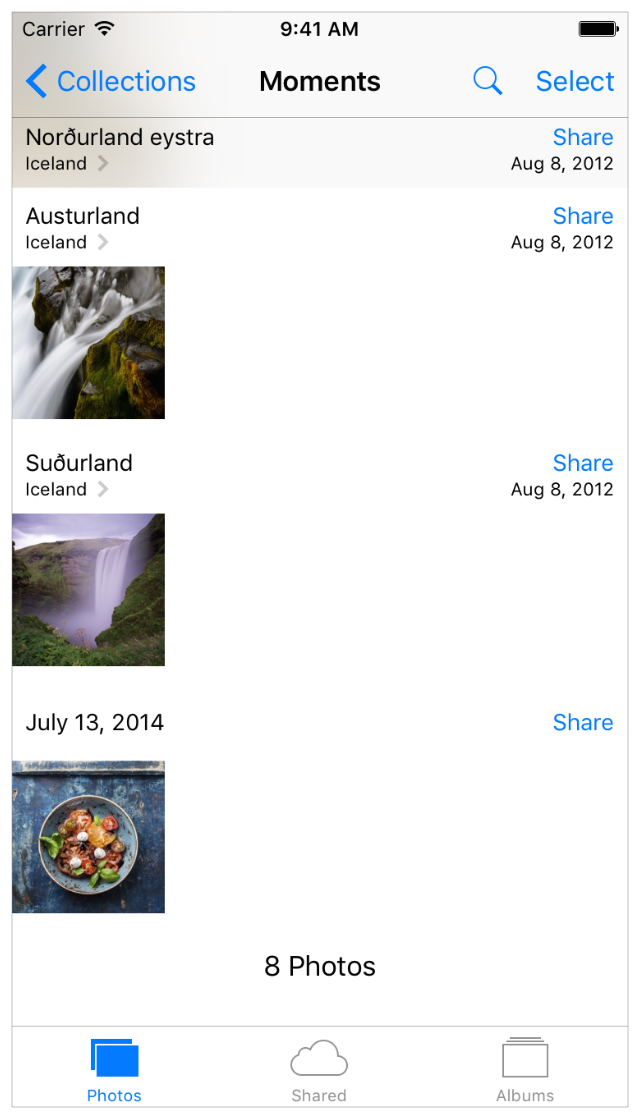
// swift 3





If you look through the photos available in Simulator, you’ll notice that it doesn’t include any photos of food. You can add your own images directly into Simulator to test the FoodTracker app with appropriate sample content. You can find a sample image within the Images/ folder of the downloadable file at the end of this lesson, or use your own image.

**To add images to Simulator**

1. If necessary, run your app in Simulator.
2. On your computer, select the images you want to add.
3. Drag and drop the images into Simulator.
4. 
5. Simulator opens the Photos app and shows the images you added.
6. 

*Checkpoint*: Run your app. You should be able to tap the image view to pull up an image picker. Open Camera Roll, and click one of the images you added to Simulator to select it and set it as the image in the image view.

