



UNIVERSIDAD PONTIFICIA DE SALAMANCA

Asignatura: Programación de Apps

Tema: Aplicación en IOS

Mail: acamarara.inf@upsa.es

Universidad: UPSA

Alumno: Ángel Cámara Ramos

La aplicación creada es un recetario donde se te muestran en diferentes tablas las diferentes comidas, con su nacionalidad de proveniencia, además de una foto de el lugar de origen y una foto de la propia receta.

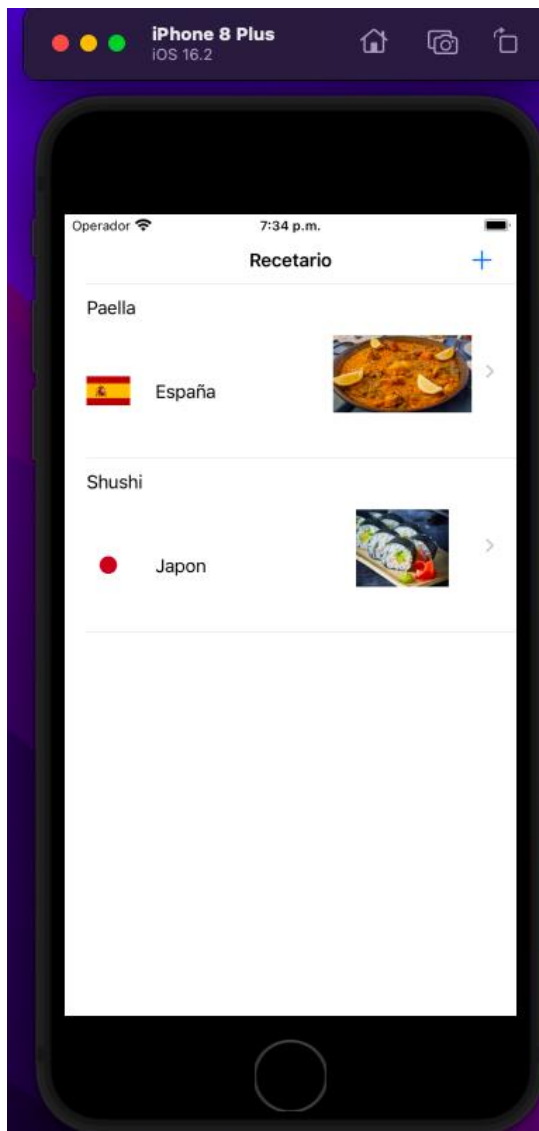
Desde aquí podemos hacer tres cosas:

- 1- Podemos Ver mas detalles de esa receta pulsando sobre la celda
- 2- Podemos eliminar una receta del listado arrastrando la celda hacia la izquierda
- 3- Podemos añadir otra receta pulsando en el “+” que tenemos en la Barra Superior, al lado de “Recetario”.

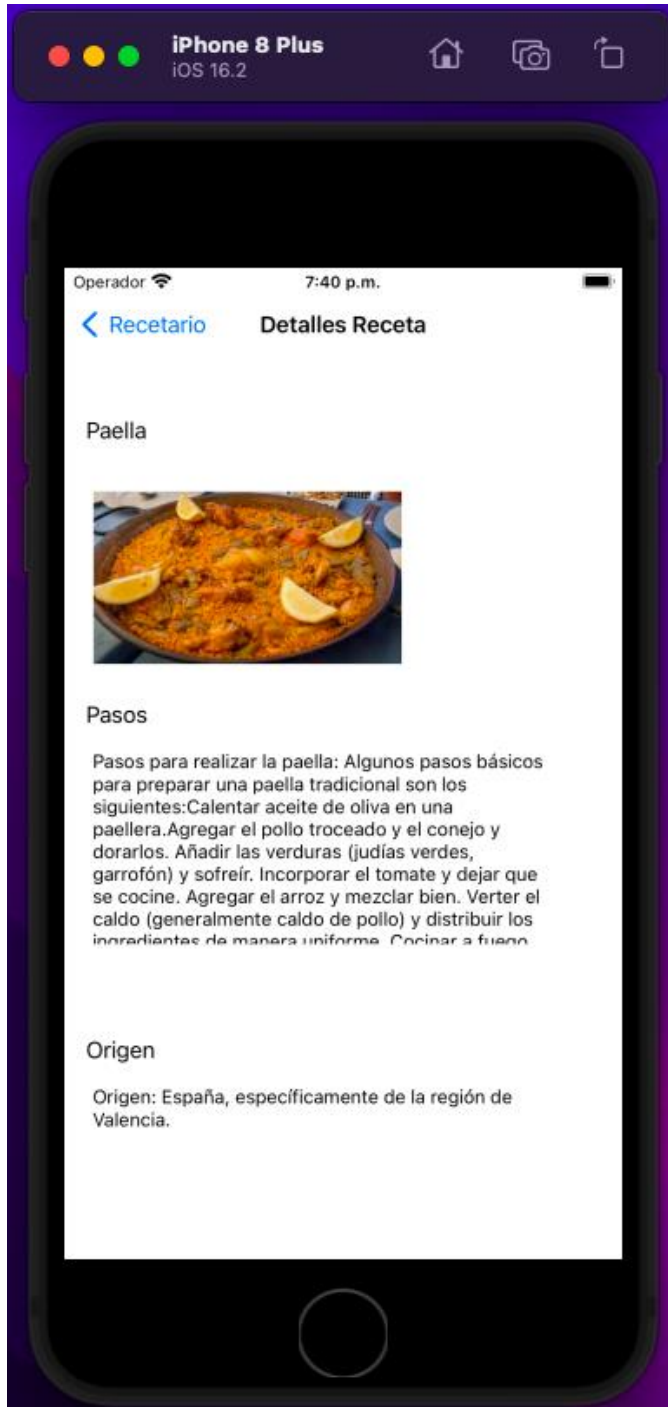
A continuación, te muestro el funcionamiento.

Ejecución de la aplicación por pasos

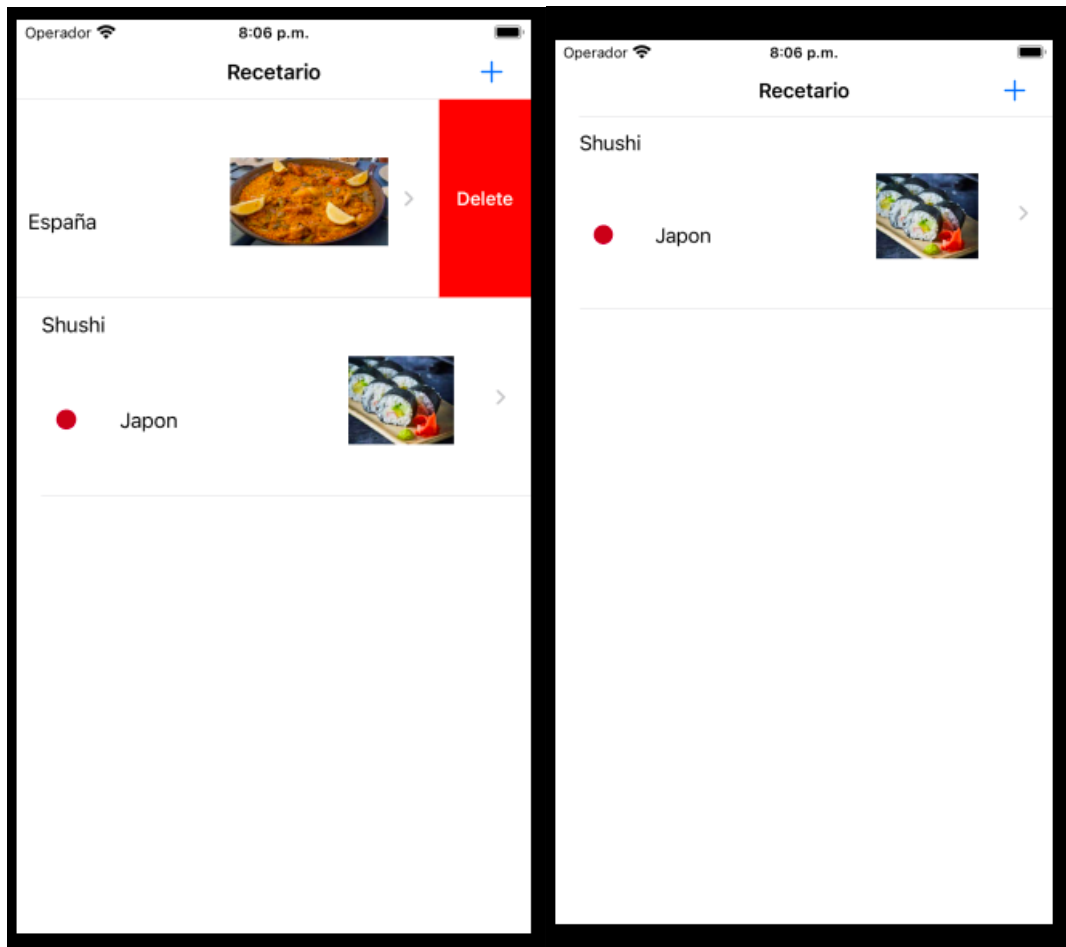
- 1- Pantalla de Inicio



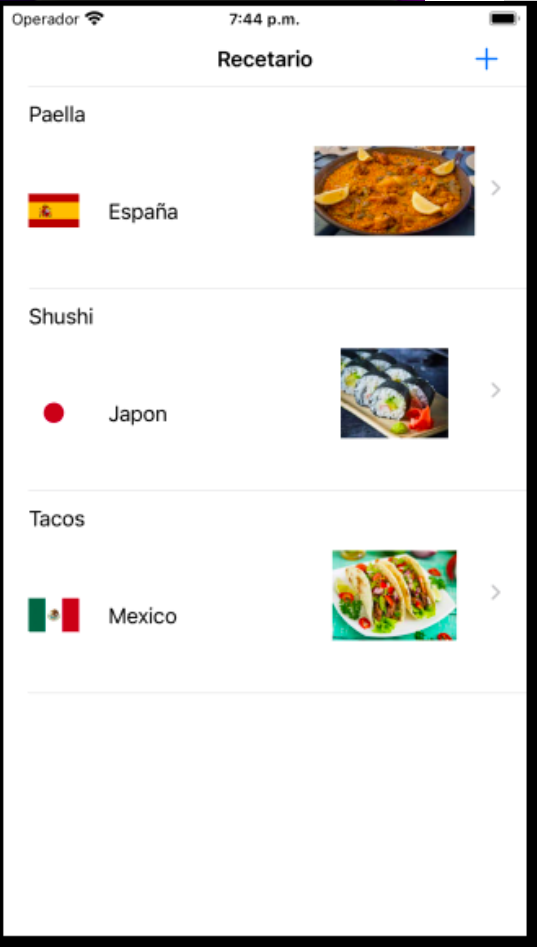
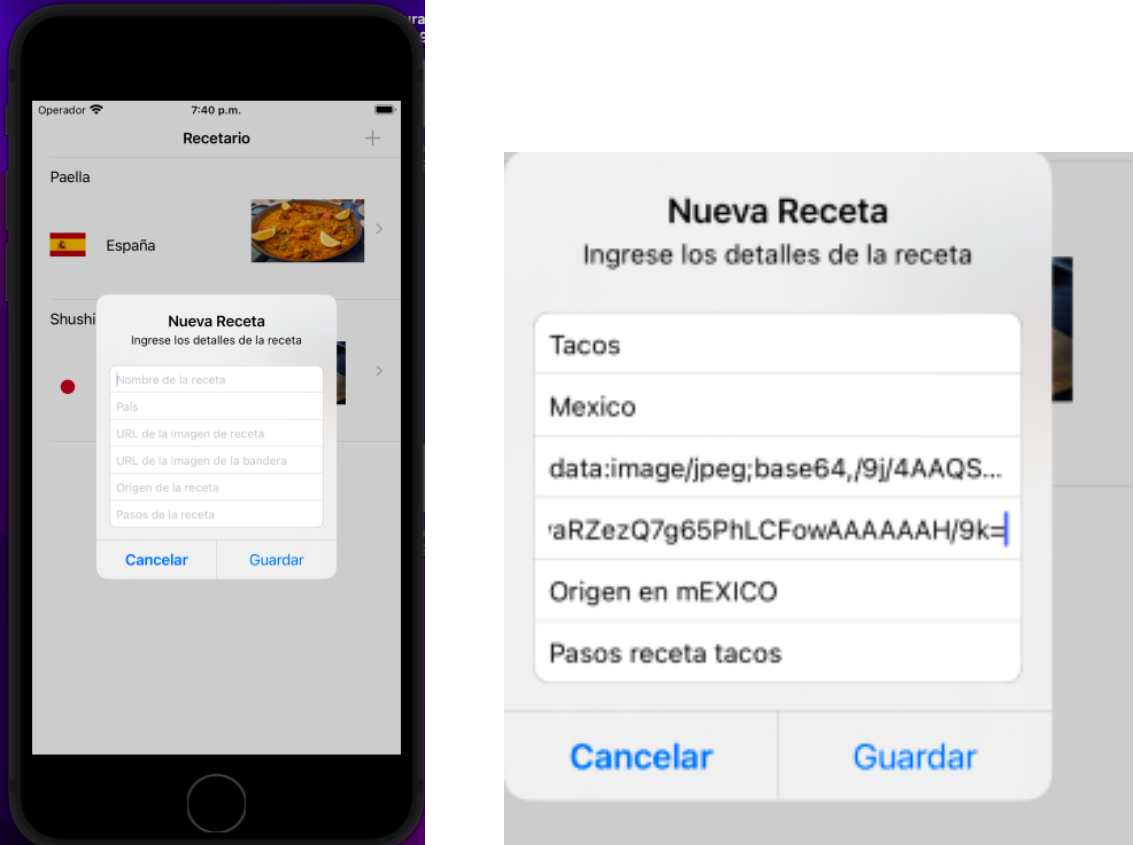
2- Selección de una de las recetas/celdas mediante click



3-Después de volver a la pagina principal, vamos a eliminar una celda/receta



4-Añadimos una nueva receta pulsando en “+”



5- Podemos ver las recetas ampliadas también de las que hemos añadido

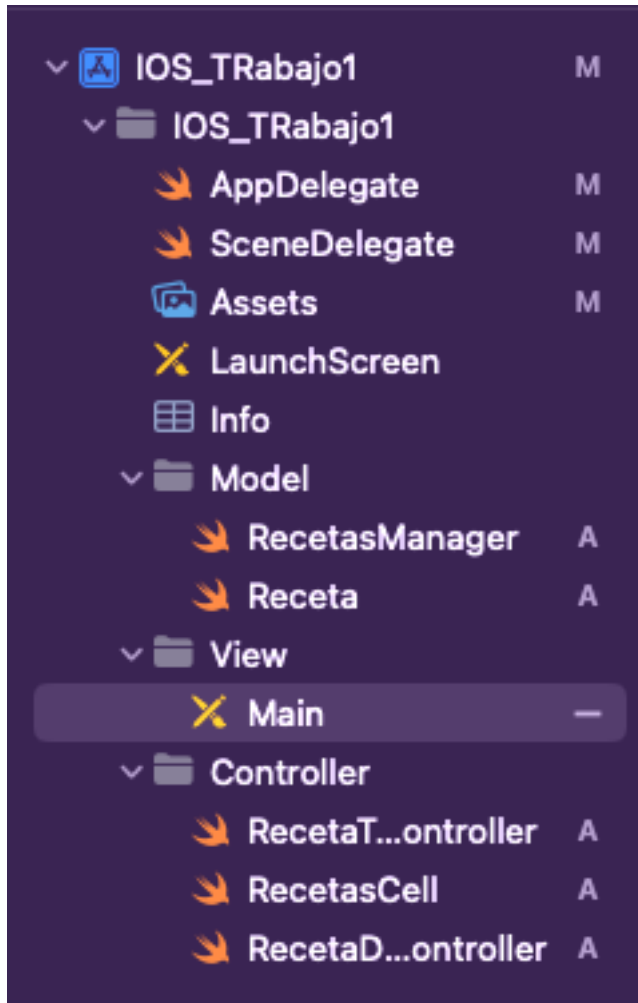


FIN DEL EJEMPLO

Clases empleadas

He dividido las clases usadas entorno a MODELO – VISTA – CONTROLADOR, para diferenciar claramente que clase pertenecía a cual.

En los Assets metí las banderas de España y Japón, así como la Paella y el Shushi que viste en la implementación.

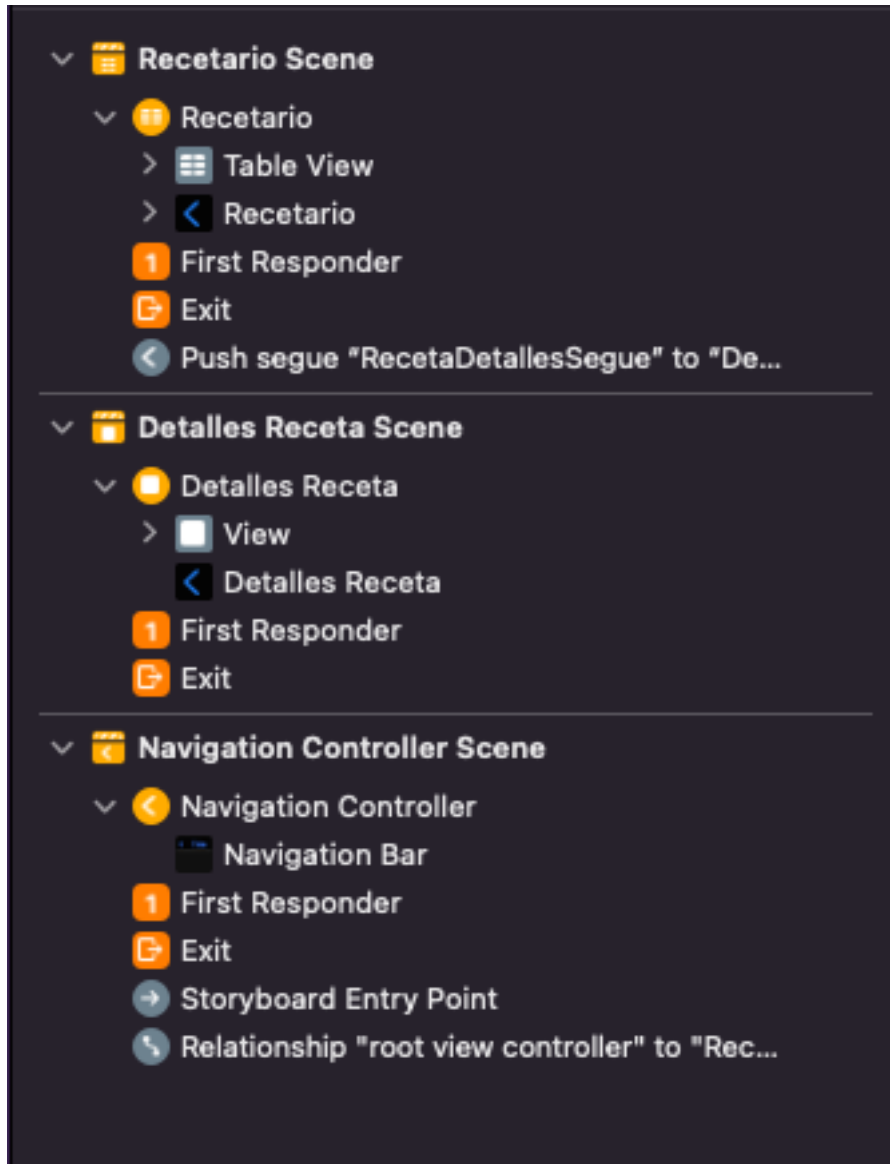


EXPLICACIONES DE LAS CLASES

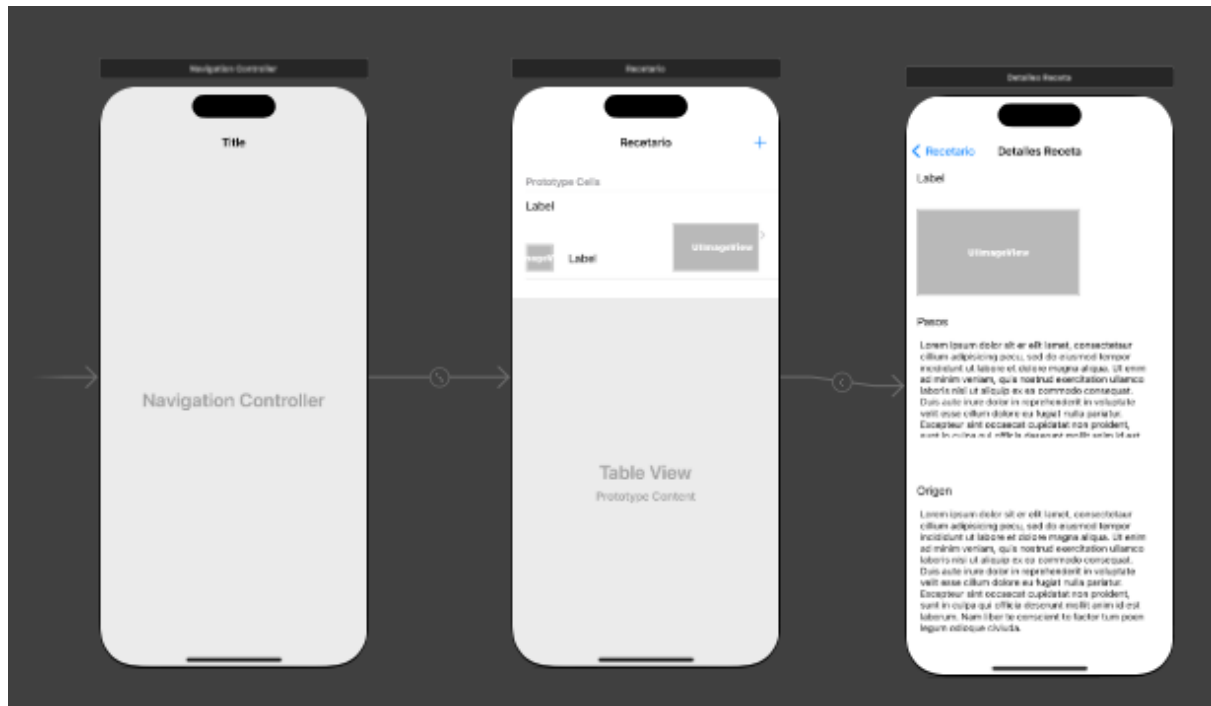
Vista

(Main):

Consta de Un Navigation controller, una TableViewcontroller (que contiene una CellView) y un ViewController.



Aquí podemos ver la estructura del programa, la relación para ver la ultima vista se produce mediante un segue mientras que los backs en la navegación se producen gracias a la navigation bar.



Modelo

(Receta):

Creación de variables y constructor.

```
7
8 import Foundation
9 import UIKit
10
11 class Receta {
12     var nombre: String
13     var pais: String
14     var imagenReceta: UIImage
15     var imagenBandera: UIImage
16     var origenReceta: String
17     var pasosReceta: String
18
19     init(nombre: String, pais: String, imagenReceta: UIImage, imagenBandera: UIImage, origenReceta: String, pasosReceta: String) {
20         self.nombre = nombre
21         self.pais = pais
22         self.imagenReceta = imagenReceta
23         self.imagenBandera = imagenBandera
24         self.origenReceta = origenReceta
25         self.pasosReceta = pasosReceta
26     }
27 }
28
29
```

(RecetasManager):

Creación del “Repositorio” donde se van a almacenar las recetas, inicialización de dos ejemplos de receta, funciones para cargar y guardar las recetas.

```
7
8 import Foundation
9 import UIKit
10
11
12 class RecetasManager
13 {
14     static let sharedInstance = RecetasManager()
15
16     var recetas: [Receta] = []
17
18     private init() {
19         // Inicialización de datos de ejemplo
20         recetas.append(Receta(nombre: "Paella", pais: "España", imagenReceta: UIImage(named: "Paella"), imagenBandera: UIImage(named: "banderaEspaña"), origenReceta: "Origen: España, específicamente de la región de Valencia.", pasosReceta: "Pasos para realizar la paella: Algunos pasos básicos para preparar una paella tradicional son los siguientes: Calentar aceite de oliva en una paellera. Agregar el pollo troceado y el conejo y dorarlos. Añadir las verduras (judías verdes, garrofón) y sofreír. Incorporar el tomate y dejar que se cocine. Agregar el arroz y mezclar bien. Verter el caldo (generalmente caldo de pollo) y distribuir los ingredientes de manera uniforme. Cocinar a fuego medio-alto durante unos minutos y luego reducir el fuego. Distribuir los mariscos, como gambas y mejillones, sobre la paella. Cocinar hasta que el arroz esté tierno y el caldo se haya absorbido. Dejar reposar unos minutos antes de servir."))
21
22         recetas.append(Receta(nombre: "Shushi", pais: "Japon", imagenReceta: UIImage(named: "Shushi"), imagenBandera: UIImage(named: "banderaJapon"), origenReceta: "Origen: Japón.", pasosReceta: "Pasos para realizar el sushi: A continuación, se mencionan algunos pasos generales para hacer sushi: Cocinar el arroz para sushi según las instrucciones del fabricante. Mientras el arroz se enfría, preparar los ingredientes, como pescado crudo (sashimi), mariscos, verduras y condimentos. Cortar el pescado en rebanadas delgadas y preparar otros ingredientes, como pepino, aguacate o zanahoria, en tiras delgadas. Extender una esterilla de bambú para sushi (makisu) y colocar una hoja de alga nori sobre ella. Humedecer las manos con agua para evitar que el arroz se pegue y tomar una porción de arroz cocido. Extender el arroz uniformemente sobre el alga nori, dejando un borde vacío en la parte superior. Colocar los ingredientes preparados en el centro del arroz. Enrollar la esterilla de bambú, presionando suavemente para asegurar el sushi. Repetir el proceso para hacer más rollos de sushi con diferentes ingredientes. Una vez hechos los rollos, cortarlos en trozos más pequeños con un cuchillo afilado. Servir el sushi con salsa de soja, wasabi y jengibre encurtido (gari)."))
23
24     }
25
26
27     // Guardar Recetas
28     // Guardar Recetas
29     func save() {
30         let encodedData = try? NSKeyedArchiver.archivedData(withRootObject: recetas, requiringSecureCoding: false)
31         UserDefaults.standard.set(encodedData, forKey: "recetas")
32     }
33
34     // Cargar Recetas
35     func load() {
36         if let encodedData = UserDefaults.standard.object(forKey: "recetas") as? Data {
37             if let savedRecetas = try? NSKeyedUnarchiver.unarchiveTopLevelObjectWithData(encodedData) as? [Receta] {
38                 recetas = savedRecetas
39             }
40         }
41     }
42 }
43
44
```

Controlador

(RecetasCell):

Añadimos los Outlets entre RecetasCell y la vista

```
7
8 import UIKit
9
10 class RecetasCell: UITableViewCell {
11
12     //creamos outlets
13
14
15     //Para imgs
16     @IBOutlet weak var imgBandera: UIImageView!
17     @IBOutlet weak var imgReceta: UIImageView!
18     //para labels
19
20     @IBOutlet weak var labelPais: UILabel!
21     @IBOutlet weak var labelReceta: UILabel!
22 }
23
```

(RecetasTableViewController):

```
1 import UIKit
2
3 class RecetaTableViewController: UITableViewController {
4     var recetasManager = RecetasManager.sharedInstance
5
6     override func viewDidLoad() {
7         super.viewDidLoad()
8         recetasManager.load()
9     }
10
11     // MARK: - Table view data source
12
13     override func numberOfSections(in tableView: UITableView) -> Int {
14         return 1
15     }
16
17     override func tableView(_ tableView: UITableView, heightForRowAt indexPath: IndexPath) -> CGFloat {
18         return 160.0
19     }
20
21     override func tableView(_ tableView: UITableView, numberOfRowsInSection section: Int) -> Int {
22         return recetasManager.recetas.count
23     }
24
25     override func tableView(_ tableView: UITableView, cellForRowAt indexPath: IndexPath) -> UITableViewCell {
26         let cell = tableView.dequeueReusableCell(withIdentifier: "reuseIdentifier", for: indexPath) as! RecetasCell
27
28         let receta = recetasManager.recetas[indexPath.row]
29
30         cell.labelReceta.text = receta.nombre
31         cell.labelPais.text = receta.pais
32         cell.imgReceta.image = receta.imagenReceta
33         cell.imgBandera.image = receta.imagenBandera
34
35         return cell
36     }
37
38     // MARK: - Navigation
39
40     override func prepare(for segue: UIStoryboardSegue, sender: Any?) {
41         if segue.identifier == "RecetaDetallesSegue",
42             let detallesViewController = segue.destination as? RecetaDetallesViewController,
43             let indexPath = tableView.indexPathForSelectedRow {
44
45             let receta = recetasManager.recetas[indexPath.row]
46             detallesViewController.receta = receta
47         }
48     }
49 }
```

```

100 // MARK: - Actions
101
102 @IBAction func addReceta(_ sender: Any) {
103     let alertController = UIAlertController(title: "Nueva Receta", message: "Ingrese los detalles de la receta", preferredStyle: .alert)
104
105     alertController.addTextField { (textField) in
106         textField.placeholder = "Nombre de la receta"
107     }
108     alertController.addTextField { (textField) in
109         textField.placeholder = "País"
110     }
111     alertController.addTextField { (textField) in
112         textField.placeholder = "URL de la imagen de receta"
113     }
114     alertController.addTextField { (textField) in
115         textField.placeholder = "URL de la imagen de la bandera"
116     }
117     alertController.addTextField { (textField) in
118         textField.placeholder = "Origen de la receta"
119     }
120     alertController.addTextField { (textField) in
121         textField.placeholder = "Pasos de la receta"
122     }
123
124     let alertAction = UIAlertAction(title: "Guardar", style: .default) { (_) in
125         if let nombreReceta = alertController.textFields?[0].text,
126            let paisReceta = alertController.textFields?[1].text,
127            let urlImagenReceta = alertController.textFields?[2].text,
128            let urlImagenBandera = alertController.textFields?[3].text,
129            let origenReceta = alertController.textFields?[4].text,
130            let pasosReceta = alertController.textFields?[5].text {
131
132             // Cargar las imágenes desde las URL proporcionadas
133             if let imagenRecetaURL = URL(string: urlImagenReceta),
134                let imagenBanderaURL = URL(string: urlImagenBandera),
135                let imagenRecetaData = try? Data(contentsOf: imagenRecetaURL),
136                let imagenBanderaData = try? Data(contentsOf: imagenBanderaURL),
137                let imagenReceta = UIImage(data: imagenRecetaData),
138                let imagenBandera = UIImage(data: imagenBanderaData) {
139
140                 let nuevaReceta = Receta(nombre: nombreReceta, pais: paisReceta, imagenReceta: imagenReceta, imagenBandera: imagenBandera, origenReceta: origenReceta,
141                                         pasosReceta: pasosReceta)
142
143                 self.recetasManager.recetas.append(nuevaReceta)
144                 self.tableView.reloadData()
145             } else {
146                 print("Error al cargar las imágenes desde las URL proporcionadas.")
147             }
148         }
149
150         let cancelAction = UIAlertAction(title: "Cancelar", style: .cancel, handler: nil)
151
152         alertController.addAction(alertAction)
153         alertController.addAction(cancelAction)
154
155         present(alertController, animated: true, completion: nil)
156     }
157
158     override func tableView(_ tableView: UITableView, commit editingStyle: UITableViewCell.EditingStyle, forRowAt indexPath: IndexPath) {
159         if editingStyle == .delete {
160             // Eliminar el elemento de la fuente de datos
161             recetasManager.recetas.remove(at: indexPath.row)
162
163             // Eliminar la celda de la tabla
164             tableView.deleteRows(at: [indexPath], with: .fade)
165         }
166     }
167
168 // MARK: - View Lifecycle
169
170 override func viewWillAppear(_ animated: Bool) {
171     super.viewWillAppear(animated)
172     tableView.reloadData()
173 }
174 }
175

```

Aquí tenemos varias funciones, empezando desde arriba tenemos:

- 1- Creación de Singleton, con sharedInstance
- 2- Hago un load cada vez que carga la vista
- 3- Devolvemos solo 1 numero de sección
- 4- Aumentamos la altura a 160, para que entre la celda completa
- 5- Contamos las recetas / celdas necesarias
- 6- Otorgamos los valores correspondientes a los labels y demás elementos de cada celda para mostrarlos en esta TableView
- 7- Unimos mediante un Segue ("RecetaDetallesSegue") un esta clase con RecetaDetalles...
- 8- Creamos una receta, para ello pasamos las variables necesarias al usuario mediante una alerta con varios campos a rellenar, la nueva receta se añade mediante append y se vuelve a cargar la vista con reload
- 9- Creamos una función que nos permita eliminar recetas, mediante remove.
- 10- Por último tenemos viewWillAppear que hace que cada vez que aparezca la pantalla, esta recargue sus elementos.

(RecetaDetallesViewController):

```
6 //
7
8 import UIKit
9
10 class RecetaDetallesViewController: UIViewController {
11
12     var receta: Receta?
13
14     override func viewDidLoad() {
15         super.viewDidLoad()
16
17
18
19
20         if let receta = receta {
21             labelNombreReceta.text = receta.nombre
22             imgReceta.image = receta.imagenReceta
23             textViewPasos.text = receta.pasosReceta
24             textViewOrigen.text = receta.origenReceta
25         }
26
27     }
28
29
30     @IBOutlet weak var labelNombreReceta: UILabel!
31
32     @IBOutlet weak var imgReceta: UIImageView!
33
34     @IBOutlet weak var textViewPasos: UITextView!
35
36     @IBOutlet weak var textViewOrigen: UITextView!
37
38
39
40
41
42
43 }
44
```

Esta clase contiene los Outlets de la información añadida de las recetas así como la parte de la Receta necesaria para su uso.

A mejorar:

Los siguientes pasos que daría para mejorar la app serían:

- 1- Añadir constraints para que pudiera verlo en horizontal o en otros dispositivos diferentes (mi implementación ha sido en iPhone 8 pro)
- 2- Añadir un scroll a la descripción y los pasos de la vista de detalles, así en caso de que no entre todo el texto el usuario podría verlo con dicho scroll
- 3- Una opción de modificar recetas
- 4- Adición de persistencia a las recetas, para que no se pierdan al cerrar la app

FIN