

## Guía práctica N5:

### Uso de tableview y manejo de elementos personalizados

#### Objetivos:

En esta sesión tiene los siguientes objetivos.

- I. Crear clases y estructuras, uso de arreglos
- II. Utilizar la vista UITableView para cargar datos de tipo lista
- III. Consumir servicios utilizando tareas asincronas



Continuar con el proyecto trabajado en la clase anterior: Semana 05

#### Paso 01: Ingresar a XCODE

Utilizar el dock de mac





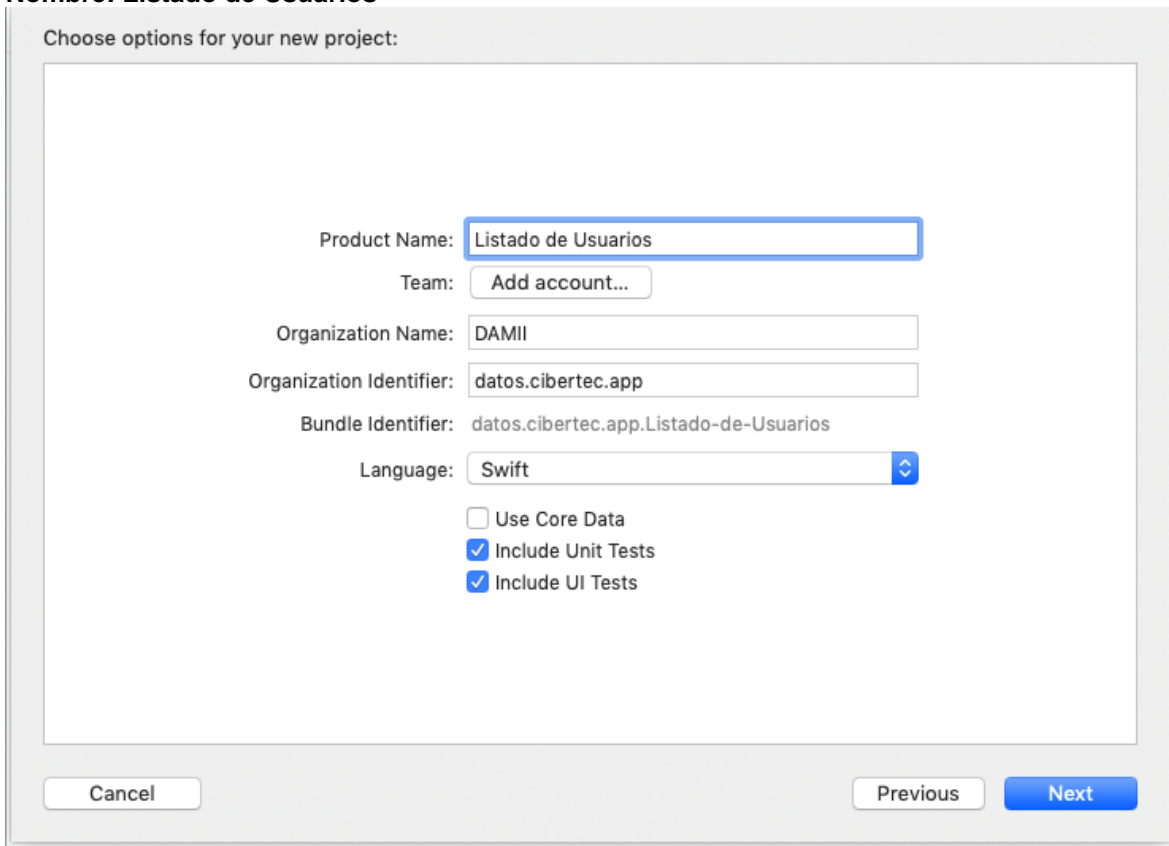
Abrir el proyecto descargado listado de equipos

**Nombre del proyecto:** Listado de equipos

**Ubicación:** en el desktop\T6AN\Semana06

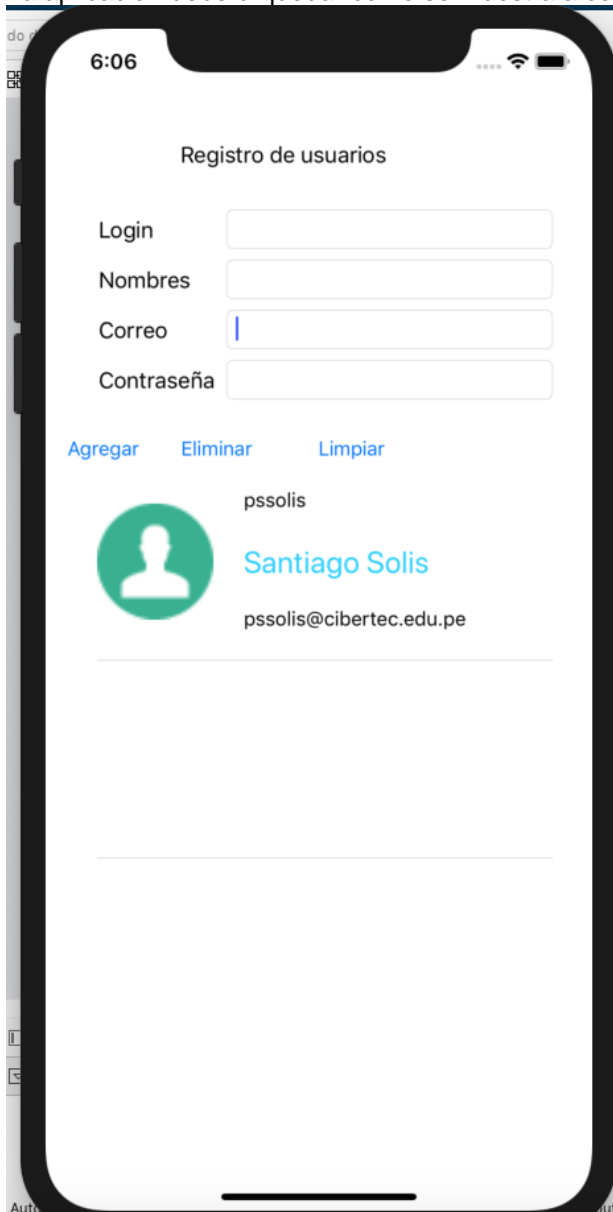
**Datos del proyecto:**

**Nombre:** Listado de Usuarios



**Registro de usuarios**

Desarrollar una aplicación que permita registrar usuarios en un tableview  
La aplicación deberá quedar como se muestra a continuación



Funcionalidades de los botones

**Agregar**

Permite agregar un nuevo usuarios con los datos ingresados todos los campos son obligatorios

**Eliminar**

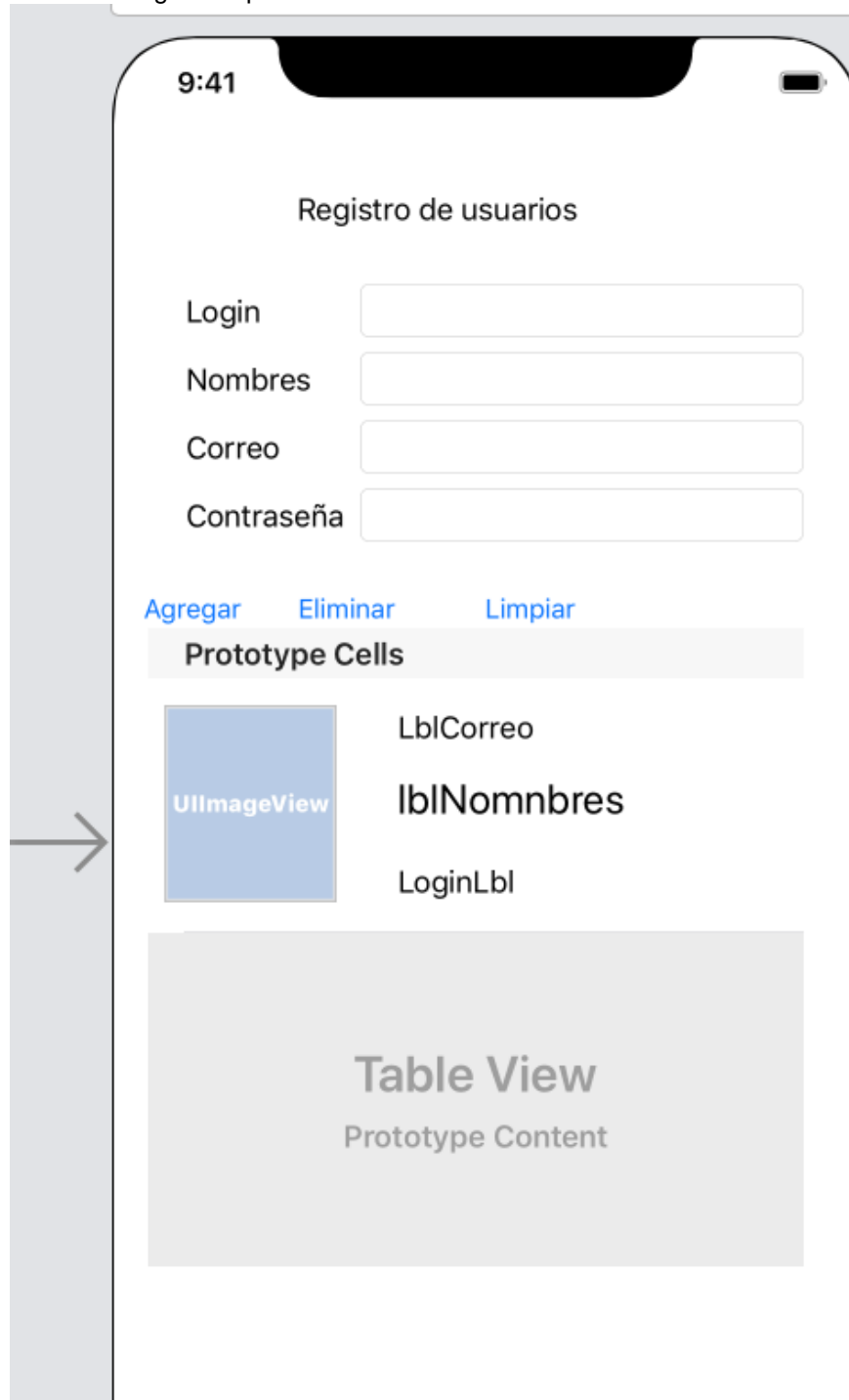
Permite eliminar un registro seleccionado de la tabla

**Limpiar**

Permite limpiar los campos de la pantalla y enviar el foco al botón del login.

**Diseño de pantalla**

Diseñar la siguiente pantalla

**Datos de la tabla:**

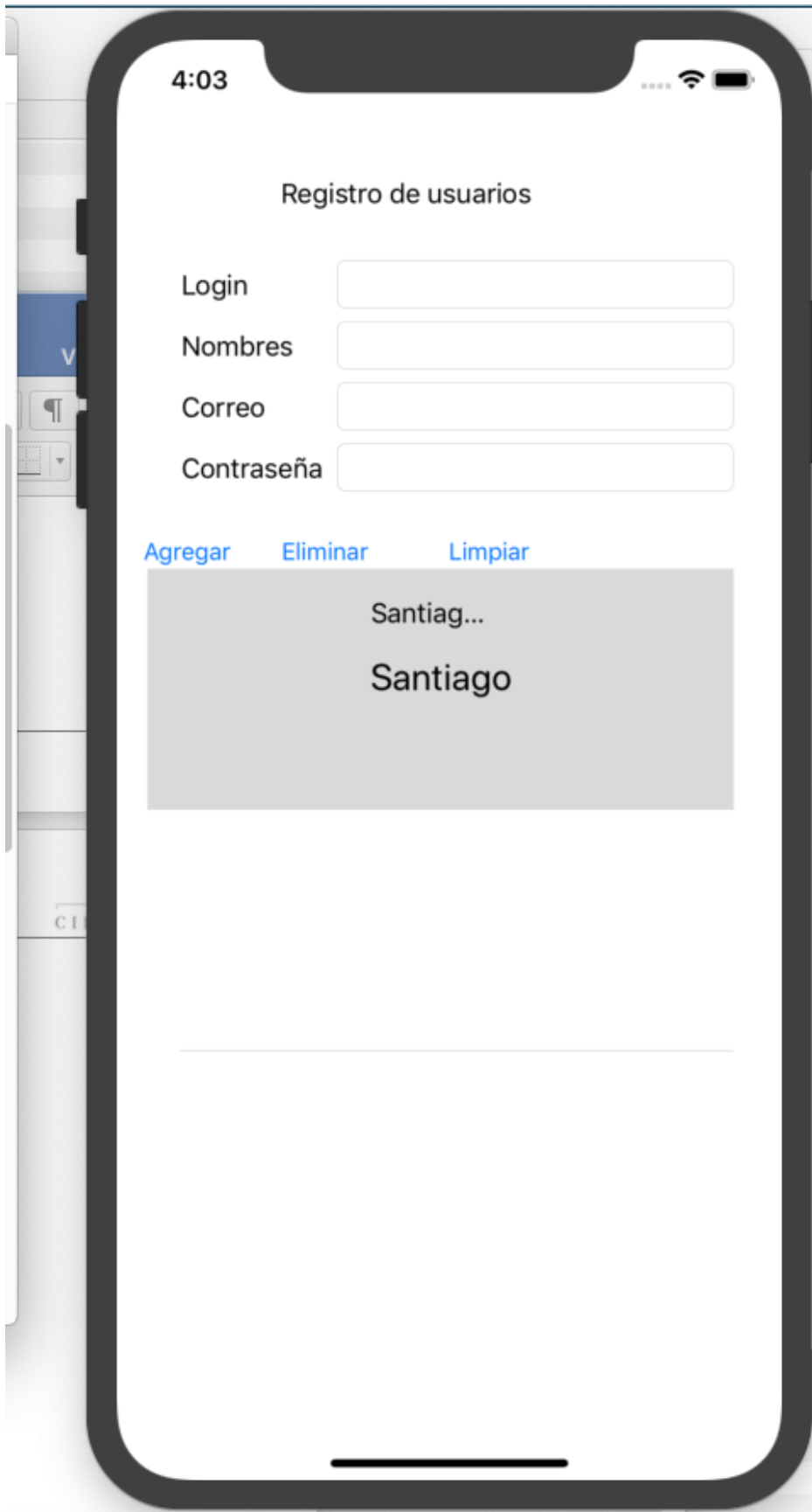
Prototype Cells=1

**Datos para la celda**

Class=CeldaUsuarioTableViewCell

Identifier=CeldaUsuario

Row Height=150



## crear la clase

Listado de Usuarios &gt; Listado de Usuarios &gt; Usuario\_BE.swift &gt; ContraseñaUsuario

```
1  //
2  //  Usuario_BE.swift
3  //  Listado de Usuarios
4  //
5  //  Created by DAMII on 10/5/19.
6  //  Copyright © 2019 DAMII. All rights reserved.
7  //
8
9  import UIKit
10
11 class Usuario_BE: NSObject {
12
13     var CódigoUsuario: integer_t
14     var LoginUsuario: String?
15     var Nombres: String?
16     var Correo: String?
17     var ContraseñaUsuario: String?
18     var Eliminado: Bool
19     var CódigoError: integer_t
20     var DescripciónError: String?
21     var MensajeError: String?
22
23     override init() {
24         self.CódigoUsuario = 0
25         self.LoginUsuario = ""
26         self.Nombres = ""
27         self.Corrreo = ""
28         self.ContraseñaUsuario = ""
29         self.Eliminado = false
30         self.CódigoError = 0
31         self.DescripciónError = ""
32         self.MensajeError = ""
33     }
34 }
35
```

**crear la clase controladora de la celda****CeldaUsuarioTableViewCell**

```
5 // Created by DAMII on 10/5/19.
6 // Copyright © 2019 DAMII. All rights reserved.
7 //
8
9 import UIKit
10
11 class CeldaUsuarioTableViewCell: UITableViewCell {
12
13     @IBOutlet weak var lblCorreo: UILabel!
14
15     @IBOutlet weak var LblNombres: UILabel!
16
17     @IBOutlet weak var lblLogin: UILabel!
18
19     override func awakeFromNib() {
20         super.awakeFromNib()
21         // Initialization code
22     }
23
24     override func setSelected(_ selected: Bool, animated: Bool) {
25         super.setSelected(selected, animated: animated)
26
27         // Configure the view for the selected state
28     }
29
30     func MostrarUsuario(pUsuario: Usuario_BE)
31     {
32         self.lblLogin.text = pUsuario.LoginUsuario
33         self.LblNombres.text = pUsuario.Nombres
34         self.lblCorreo.text = pUsuario.Correo
35     }
36
37
38 }
39
```

## Programar el viewcontroller

```

8
9  import UIKit
10
11  class ViewController: UIViewController, UITableViewDataSource, UITableViewDelegate {
12
13      @IBOutlet weak var tfLogin: UITextField!
14      @IBOutlet weak var tfNombres: UITextField!
15      @IBOutlet weak var tfCorreo: UITextField!
16      @IBOutlet weak var tfContraseña: UITextField!
17      @IBOutlet weak var tblListado: UITableView!
18      var oListaUsuarios = [Usuario_BE]()
19
20      override func viewDidLoad() {
21          super.viewDidLoad()
22
23          self.tblListado.dataSource=self
24          self.tblListado.delegate=self
25          self.tblListado.rowHeight = 150
26          var objUsuario = Usuario_BE()  ⚠ Variable 'objUsuario' was never mutated; consider changing to 'let' const
27          objUsuario.CodigoUsuario = 1
28          objUsuario.Nombres = "Santiago"
29          objUsuario.Correo = "ssolis@cibertec.edu.pe"
30          oListaUsuarios.append(objUsuario)
31          // Do any additional setup after loading the view.
32      }
33
34      override func didReceiveMemoryWarning() {
35          super.didReceiveMemoryWarning()
36      }
37
38      func tableView(_ tableView: UITableView, numberOfRowsInSection section: Int) -> Int {
39          return oListaUsuarios.count
40      }
41
42      func numberOfSections(in tableView: UITableView) -> Int {
43          // #warning Incomplete implementation, return the number of sections
44          return 1
45      }
46
47      func tableView(_ tableView: UITableView, cellForRowAt indexPath: IndexPath) ->
      UITableViewCell {
48          let oCelda = tableView.dequeueReusableCell(withIdentifier: "CeldaUsuario", for:
      indexPath) as! CeldaUsuarioTableViewCell
49
50          let oregUsuario=oListaUsuarios[indexPath.row]
51          oCelda.MostrarUsuario(pUsuario: oregUsuario)
52          return oCelda
53      }
54
55  }
56
57

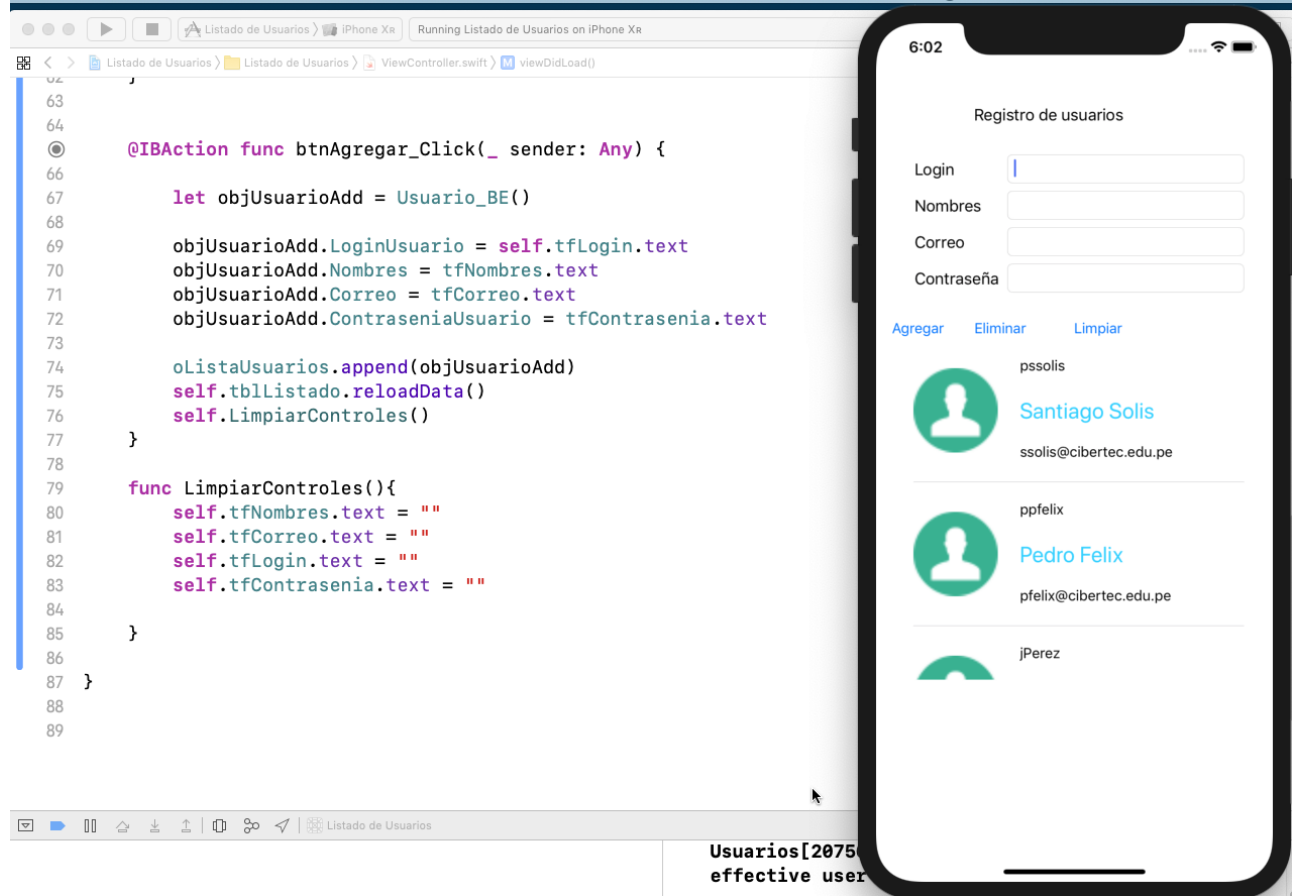
```



### Completar el diseño de los controles del tableViewCell

- Agregar la imagen y asociarlo al imageView
- No olvidarse agregar las funcionalidades de constraint layout

### Código del botón Agregar



The image shows a screenshot of the Xcode IDE. On the left, the Swift code for the `btnAgregar_Click` function is displayed, which handles the logic for adding a new user. On the right, a preview of the app's user registration screen is shown, featuring input fields for login, name, email, and password, along with buttons for adding, deleting, and clearing data. Below the preview, a list of users is visible, including 'Santiago Solis' and 'Pedro Felix'.

```
63
64
65 @IBAction func btnAgregar_Click(_ sender: Any) {
66     let objUsuarioAdd = Usuario_BE()
67
68     objUsuarioAdd.LoginUsuario = self.tfLogin.text
69     objUsuarioAdd.Nombres = tfNombres.text
70     objUsuarioAdd.Correo = tfCorreo.text
71     objUsuarioAdd.ContraseniaUsuario = tfContrasenia.text
72
73     oListaUsuarios.append(objUsuarioAdd)
74     self.tblListado.reloadData()
75     self.LimpiarControles()
76 }
77
78
79 func LimpiarControles(){
80     self.tfNombres.text = ""
81     self.tfCorreo.text = ""
82     self.tfLogin.text = ""
83     self.tfContrasenia.text = ""
84 }
85
86
87 }
88
89
```

Usuarios[2075  
effective user

Finalmente probar la aplicación