

Guía práctica N6.2:

Consumo de servicios REST

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

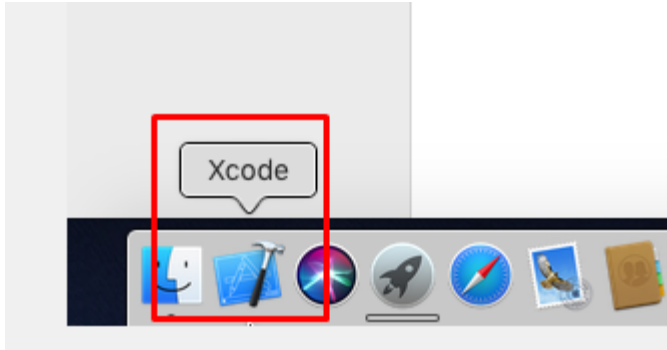


Continuar con el proyecto trabajado en la clase anterior: Semana 05

Paso 01: Ingresar a XCODE

Utilizar el dock de mac





Abrir el proyecto descargado listado de equipos

Nombre del proyecto: Control de acceso WS

Ubicación: en el desktop\T6AN\Semana07

Datos del proyecto:

Nombre: Control de Acceso

Choose options for your new project:

Product Name:	Login Usuario
Team:	Sotelo Solis Vidal (Personal Team)
Organization Name:	Santiago Solis
Organization Identifier:	cibertec.apple.com
Bundle Identifier:	cibertec.apple.com.Login-Usuario
Language:	Swift
<input type="checkbox"/> Use Core Data	
<input checked="" type="checkbox"/> Include Unit Tests	
<input checked="" type="checkbox"/> Include UI Tests	

Cancel Previous Next

Url a Utilizar:

<http://wscibertec201922.somee.com/Usuario/ListarKey?pCodigoUsuario=1>

1. Los datos de usuario se pueden listar utilizando la siguiente URL
<http://wscibertec201922.somee.com/Usuario/Lista?Nombres=prof>

2. Donde la url es :
<http://wscibertec201922.somee.com/>
3. El servicio de funcionalidades de usuario es:
Usuario
4. El método o consulta de usuario es:
Lista?Nombres=prof
5. Los parámetros de la consulta son:
Nombres=prof

Ejemplo Listar todos:

<http://wscibertec201922.somee.com/Usuario/Lista>

Servicio para validar los usuarios

<http://wscibertec201922.somee.com/Usuario/ValidarUsuario?pLogin=pssolis@cibertec.edu.pe&pContrasenia=123456789>

En el caso del desarrollo a utilizar en Swift se deberá utilizar los campos de login y clave respectivamente.

Configurar los permisos a la aplicación

Una vez creado la aplicación se deberá configurar los permisos para acceso a internet en específico al servicio a utilizar.

Login Usuario > Login Usuario > Info.plist > No Selection			
Key	Type	Value	
▼ Information Property List	Dictionary	(16 items)	
Localization native development region	String	\$(DEVELOPMENT_LANGUAGE)	
Executable file	String	\$(EXECUTABLE_NAME)	
Bundle identifier	String	\$(PRODUCT_BUNDLE_IDENTIFIER)	
InfoDictionary version	String	6.0	
Bundle name	String	\$(PRODUCT_NAME)	
Bundle OS Type code	String	APPL	
Bundle versions string, short	String	1.0	
Bundle version	String	1	
Application requires iPhone environment	Boolean	YES	
▼ App Transport Security Settings	Dictionary	(2 items)	
Allow Arbitrary Loads	Boolean	YES	
▼ Exception Domains	Dictionary	(3 items)	
NSExceptionAllowsInsecureHTTPLoads	Boolean	YES	
NSIncludesSubdomains	Boolean	YES	
http://www.wscibertec201922.somee.com	String		
Application Category	String		

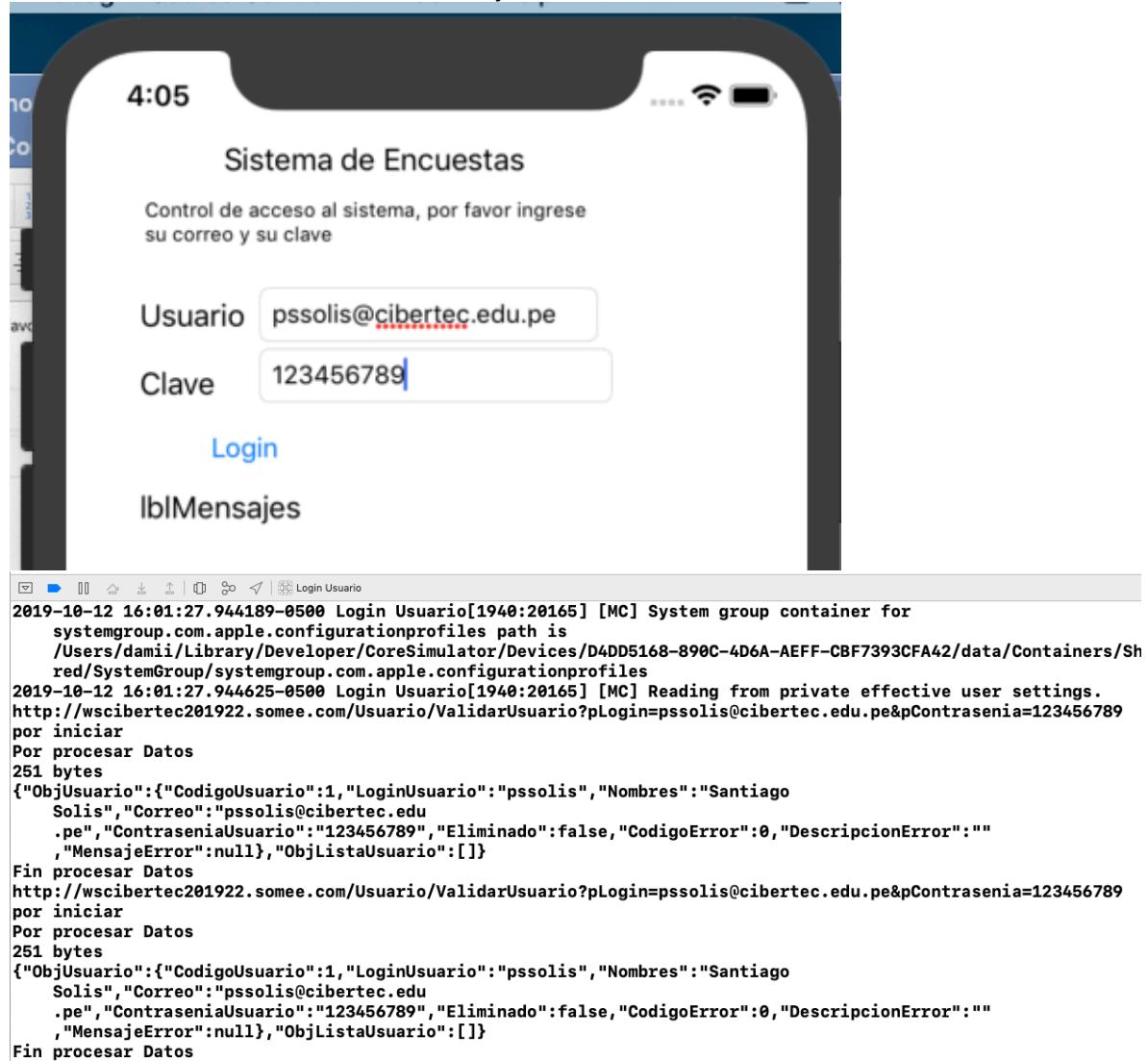
Diseñar la siguiente pantalla control de acceso**Declarar las variables**

```
8
9 import UIKit
10
11 class ViewController: UIViewController {
12     @IBOutlet weak var tfUsuario: UITextField!
13     @IBOutlet weak var tfClave: UITextField!
14     @IBOutlet weak var lblMensajes: UILabel!
15
16     override func viewDidLoad() {
17         super.viewDidLoad()
18         // Do any additional setup after loading the view, typically from a nib.
19     }
20 }
```

Programar el evento del botón

```
23  @IBAction func btnLogin_OnClick(_ sender: Any) {
24      var urlLogin="http://wscibertec201922.somee.com/Usuario/ValidarUsuario?login=" +
25          self.tfUsuario.text! + "&pContrasenia=" + self.tfClave.text!
26      print(urlLogin)
27      let urlLogueo = URL(string: urlLogin)
28      let petition = URLRequest(url: urlLogueo!)
29      let tarea = URLSession.shared.dataTask(with: petition)
30      {Datos, Respuesta, Error in
31          print("por iniciar")
32          if (Error == nil)
33          {
34              print("Por procesar Datos")
35              print(Datos ?? "Vacio")
36              let datosCadena = NSString(data: Datos!, encoding: String.Encoding.utf8.rawValue)
37              print (datosCadena!)
38              print("Fin procesar Datos")
39          }
40          else
41          {
42              print("Error")
43              print(Error ?? "Error vacio")
44              let strCadena = Error as! String
45              //self.tvMensaje.text = strCadena
46          }
47          tarea.resume()
48      }
49  }
```

Visualizar los datos como resultado de la ejecución



Leer objetos JSON

En el código del boton Login agregar en despues de la línea “39”

```

34      {
35          print("Por procesar Datos")
36          print(Datos ?? "Vacio")
37          let datosCadena = NSString(data: Datos!, encoding:
38              String.Encoding.utf8.rawValue)
39          print (datosCadena!)
40          print("Fin procesar Datos")
41      }

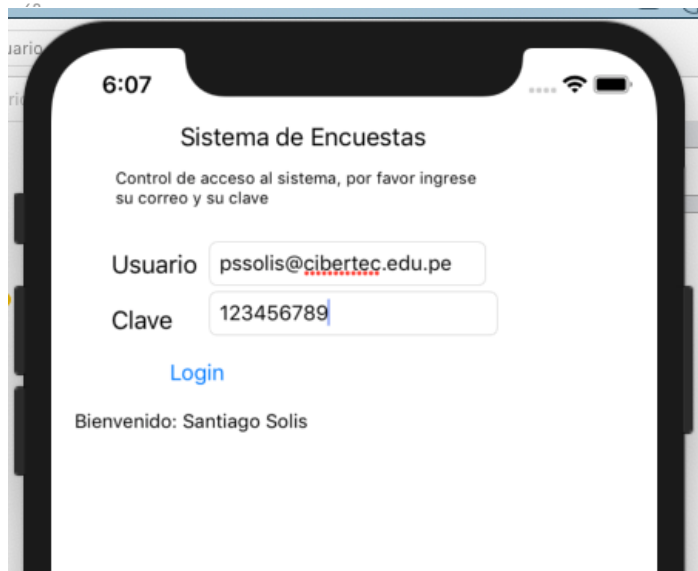
```

despues de “Fin procesar datos” Agregar el siguiente código

```

41      ///////////////////////////////////////////////////
42      print("Leer Datos JSON")
43      DispatchQueue.main.async {
44          //Inicio Completar la lectura de objetos JSON
45          let JSON = try? JSONSerialization.jsonObject(with: Datos!, options: [])
46
47          if let dictionary = JSON as? [String: Any],
48             let objUsuario = dictionary["ObjUsuario"] as? [String: Any]
49          {
50              let CodigoUsuario = objUsuario["CodigoUsuario"] as! Int
51              let Nombres = objUsuario["Nombres"] as! String
52              if (CodigoUsuario > 0)
53              {
54                  //self.NombresUsuario = Nombres
55                  self.lblMensajes.text = "Bienvenido: \(Nombres)"
56                  //self.performSegue(withIdentifier: "ListadoUsuario", sender: self)
57              }
58              else
59              {
60                  self.lblMensajes.text = "Usuario o Clave incorrecto "
61              }
62          }
63          //Fin Completar la lectura de objetos JSON
64
65      }
66      print("Fin de Leer datos JSON")
67      ///////////////////////////////////////////////////

```

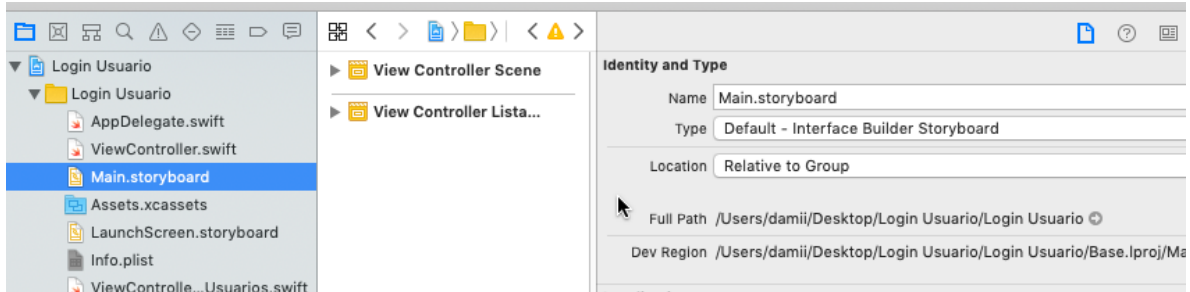


Finalmente probar la aplicación

Llamar a una segunda vista por código

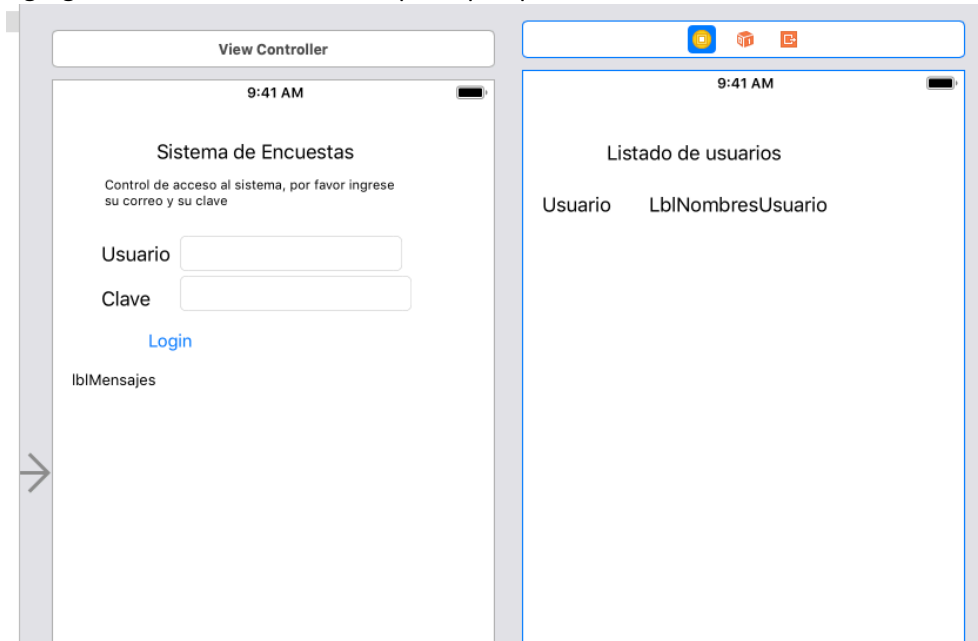
Identificar el nombre del mainstoryboard

Propiedad Name



Agregar una segunda Vista

Agregar los controles necesarios para que quede como se muestra

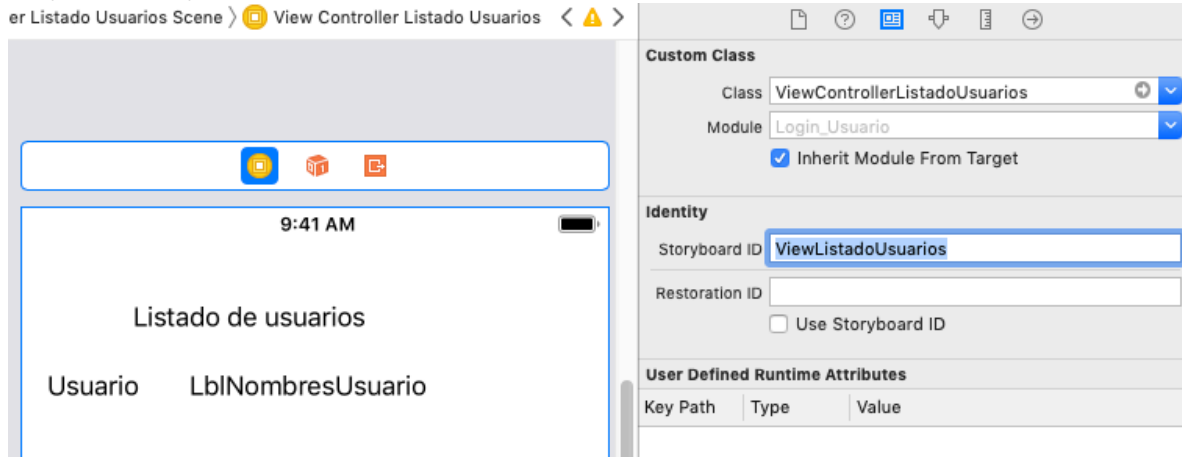


Agregar el controlador para la vista

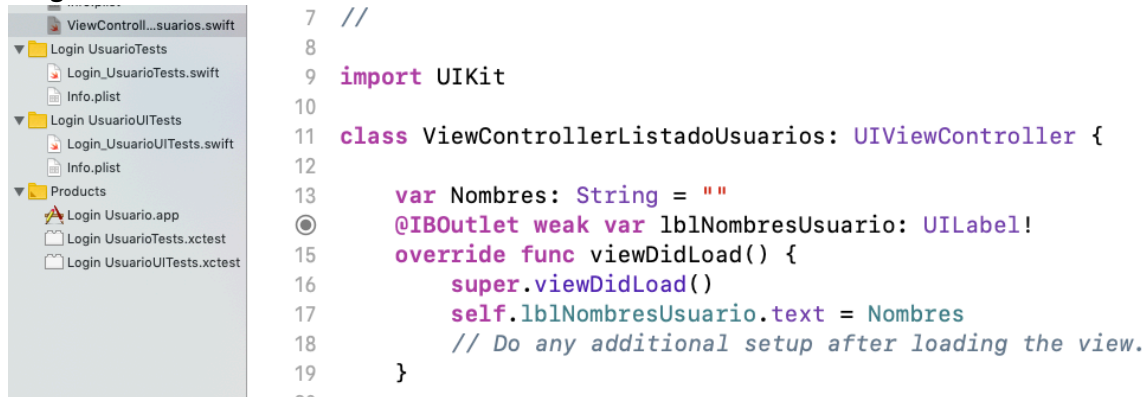
```
class ViewControllerListadoUsuarios:
    UIViewController {
```


Asociar el controlador a la vista y asignar un ID a la vista

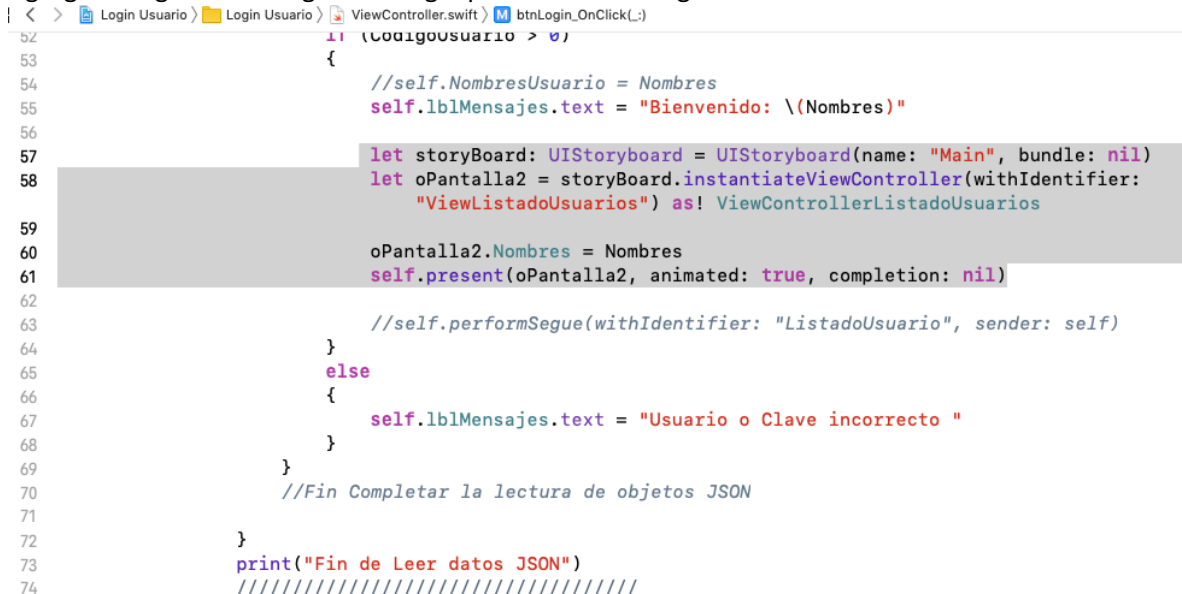
Propiedad storyboard ID



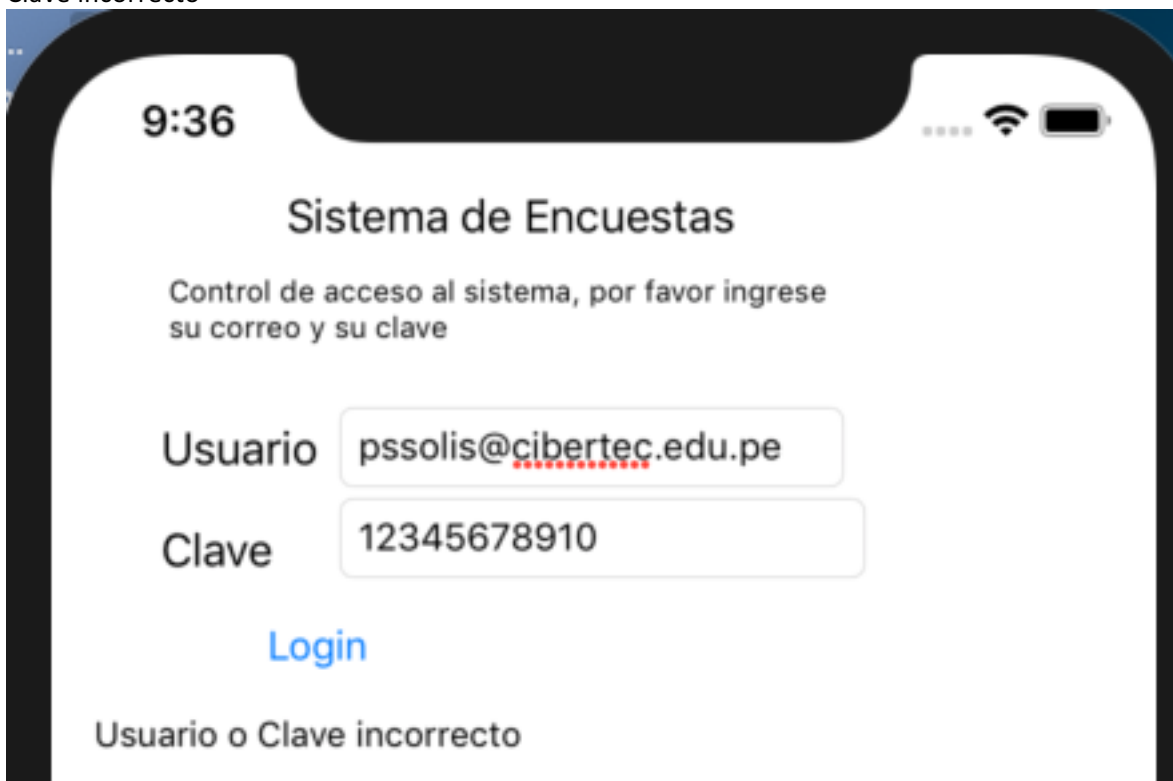
Programar el controlador creado como se muestra



Agregar el siguiente código en el login para llamara la segunda vista creada



Probar la aplicación
Clave incorrecto



Clave correcto

