



POLITÉCNICA

ETSIT
UPM

dit
UPM

Desarrollo de Apps para iOS

Navigation Controller

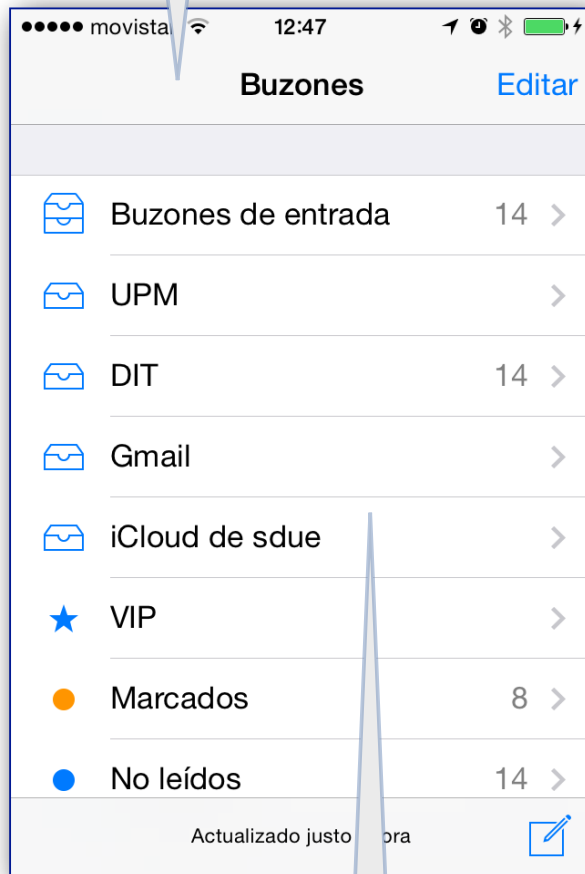
IWEB 2015-2016
Santiago Pavón

ver: 2015.10.30

Barra de título creada por UINavigationController

Propiedad **title** del VC mostrado.

Botón de retroceso al VC anterior.
Etiquetada con su title.



La vista (view) del VC actual.

Cada VC mostrado tiene su propia barra de herramientas con sus **toolbarItems**.

Funcionamiento de UINavigationController

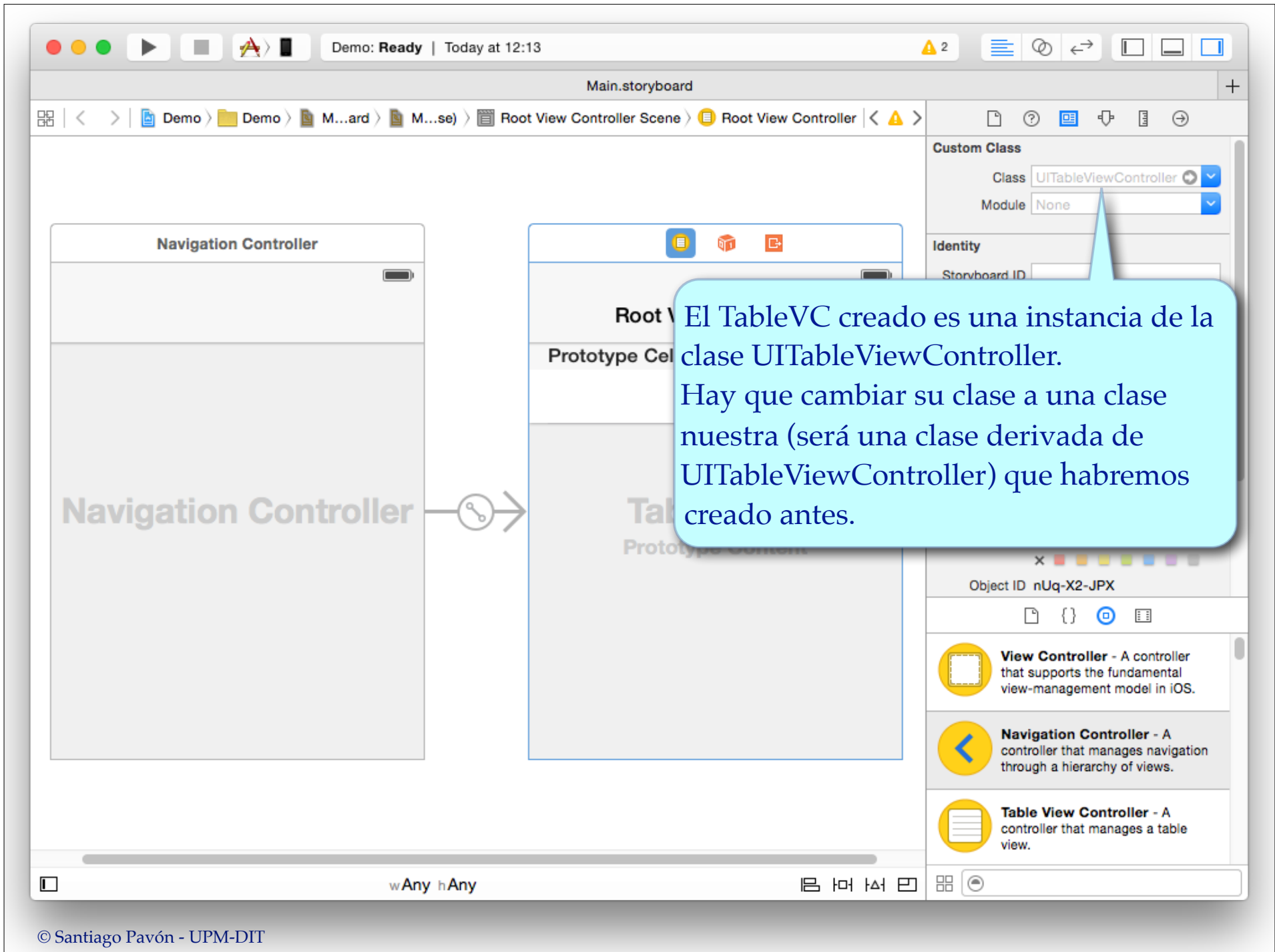
- La navegación con **UINavigationController** consiste en pasar por una serie de pantallas nuevas (View Controllers) con las que se crea una pila.
 - Para retroceder por la secuencia de pantallas por la que hemos pasado puede usarse el botón situado en la izquierda de la barra de navegación.
 - Opcionalmente, puede existir un botón en la derecha de la barra de navegación para pasar a la pantalla siguiente.
 - Programáticamente se puede saltar a la pantalla que se desee.
- Un **UINavigationController** muestra:
 - Una barra de navegación (**UINavigationController**) con:
 - el título de la pantalla actual.
 - asociado a la propiedad **title** del VC mostrado.
 - un botón (**UIBarButtonItem**) para volver a la pantalla anterior.
 - etiquetado con el título de la pantalla anterior.
 - mas botones (**UIBarButtonItem**) añadidos explícitamente.
 - La view del VC que esté actualmente en el **top** de la pila.
 - Barra de herramientas (**UIToolBar**) que muestra el contenido de la propiedad **toolbarItems** del VC actual.
 - **toolbarItems** es un array de objetos **UIBarButtonItem**.

Storyboard

- Editar los ficheros storyboard para incluir Navigation Controllers:
 - Arrastrar un objeto Navigation Controller desde la biblioteca de objetos.
 - Se crea un objeto Navigation Controller y un objeto TableVC.
 - Si queremos seguir trabajando usando el TableVC creado, tendremos que usar el inspector de identidad para indicar cuál es la clase real del TableVC.
 - También podemos borrar el TableVC creado y enganchar el Navigation Controller con algún VC ya existente.
 - Seleccionar un VC ya existente en el storyboard y ejecutar:
Menú Editor > Embed In > Navigation Controller

Arrastrar un Navigation Controller desde la biblioteca de objetos. Se crea también un objeto TableVC.

The screenshot shows the Xcode storyboard editor. On the left, a storyboard contains a **Navigation Controller** widget and a **Table View** widget. A red arrow points from the **Navigation Controller** widget to the **Table View** widget, indicating a segue. On the right, the **Attributes Inspector** is open, showing settings for the selected **Navigation Controller**. The **Simulated Metrics** section includes Size (Freeform), Orientation (Inferred), Status Bar (Inferred), Top Bar (Inferred), and Bottom Bar (Inferred). The **Navigation Controller** section includes Bar Visibility (checked), Shows Navigation Bar (checked), Shows Toolbar (unchecked), Hide Bars (unchecked), On Swipe (unchecked), On Tap (unchecked), When Keyboard Appears (unchecked), and When Vertically Compact (unchecked). The **View Controller** section includes Title (empty), Is Initial View Controller (unchecked), and Layout (checked, Adjust Scroll View Insets). The bottom of the inspector shows a list of objects: **View Controller**, **Navigation Controller**, and **Table View Controller**.

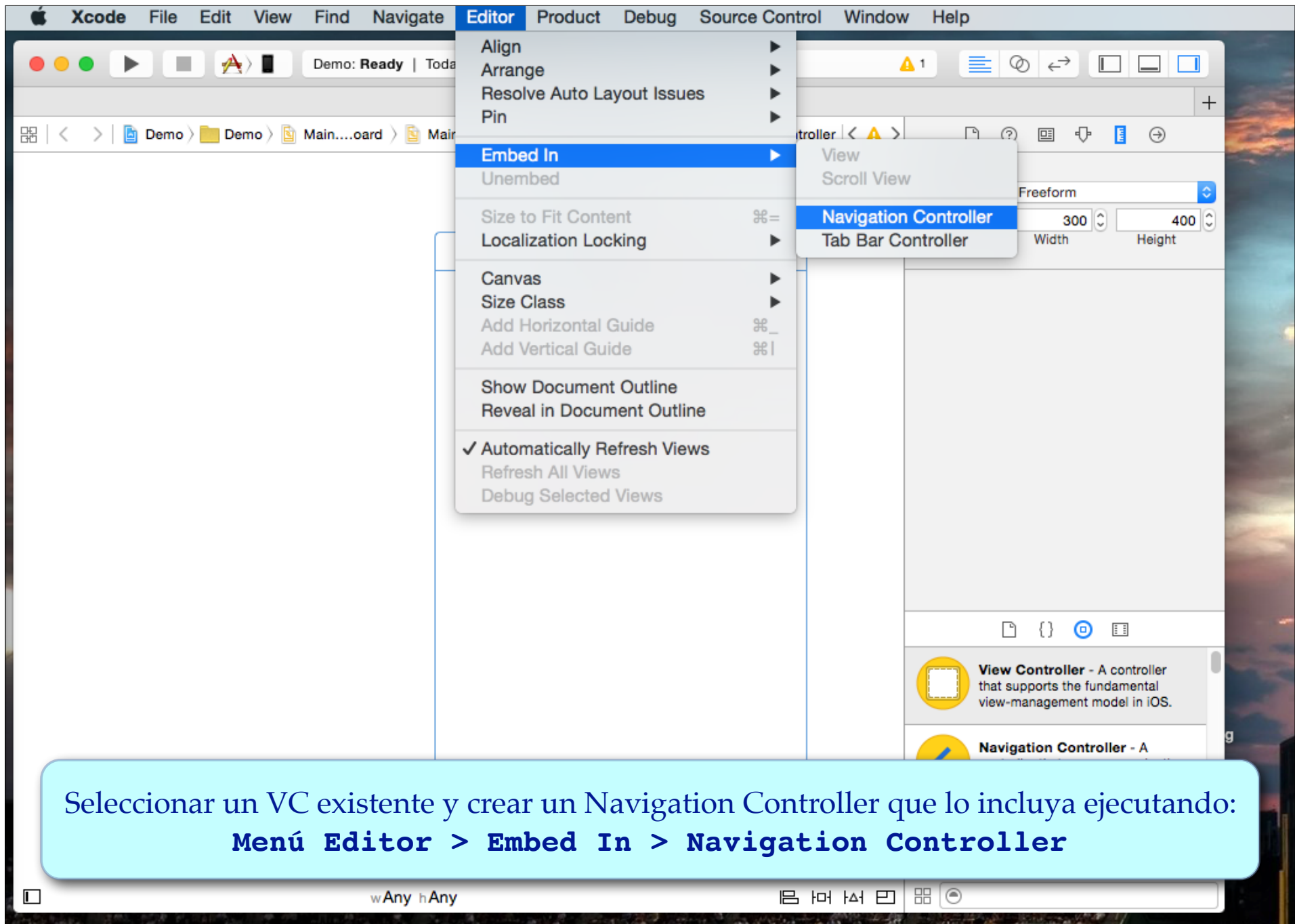


Storyboard

- Editar los ficheros storyboard para incluir Navigation Controllers:

- Arrastrar un objeto Navigation Controller desde la biblioteca de objetos.
 - Se crea un objeto Navigation Controller y un objeto TableVC.
 - Si queremos seguir trabajando usando el TableVC creado, tendremos que usar el inspector de identidad para indicar cual es la clase real del TableVC.
 - También podemos borrar el TableVC creado y enganchar el Navigation Controller con algún VC ya existente.
- Seleccionar un **VC ya existente** en el storyboard y ejecutar:

Menú Editor > Embed In > Navigation Controller



Seleccionar un VC existente y crear un Navigation Controller que lo incluya ejecutando:
Menú Editor > Embed In > Navigation Controller

The screenshot shows the Xcode storyboard editor interface. The main canvas displays a storyboard with two scenes: a **Navigation Controller** on the left and a **View Controller** on the right. An arrow with a link icon points from the Navigation Controller to the View Controller, indicating that the View Controller is embedded within the Navigation Controller. The right-hand sidebar contains the following sections:

- Simulated Metrics:** Size (Freeform), Orientation (Inferred), Status Bar (Inferred), Top Bar (Inferred), Bottom Bar (Inferred).
- Navigation Controller:** Bar Visibility (checked, Shows Navigation Bar), Shows Toolbar (unchecked), Hide Bars (On Swipe, On Tap, When Keyboard Appears, When Vertically Compact, all unchecked).
- View Controller:** Title (empty field), Is Initial View Controller (unchecked), Layout (Adjust Scroll View Insets, unchecked).

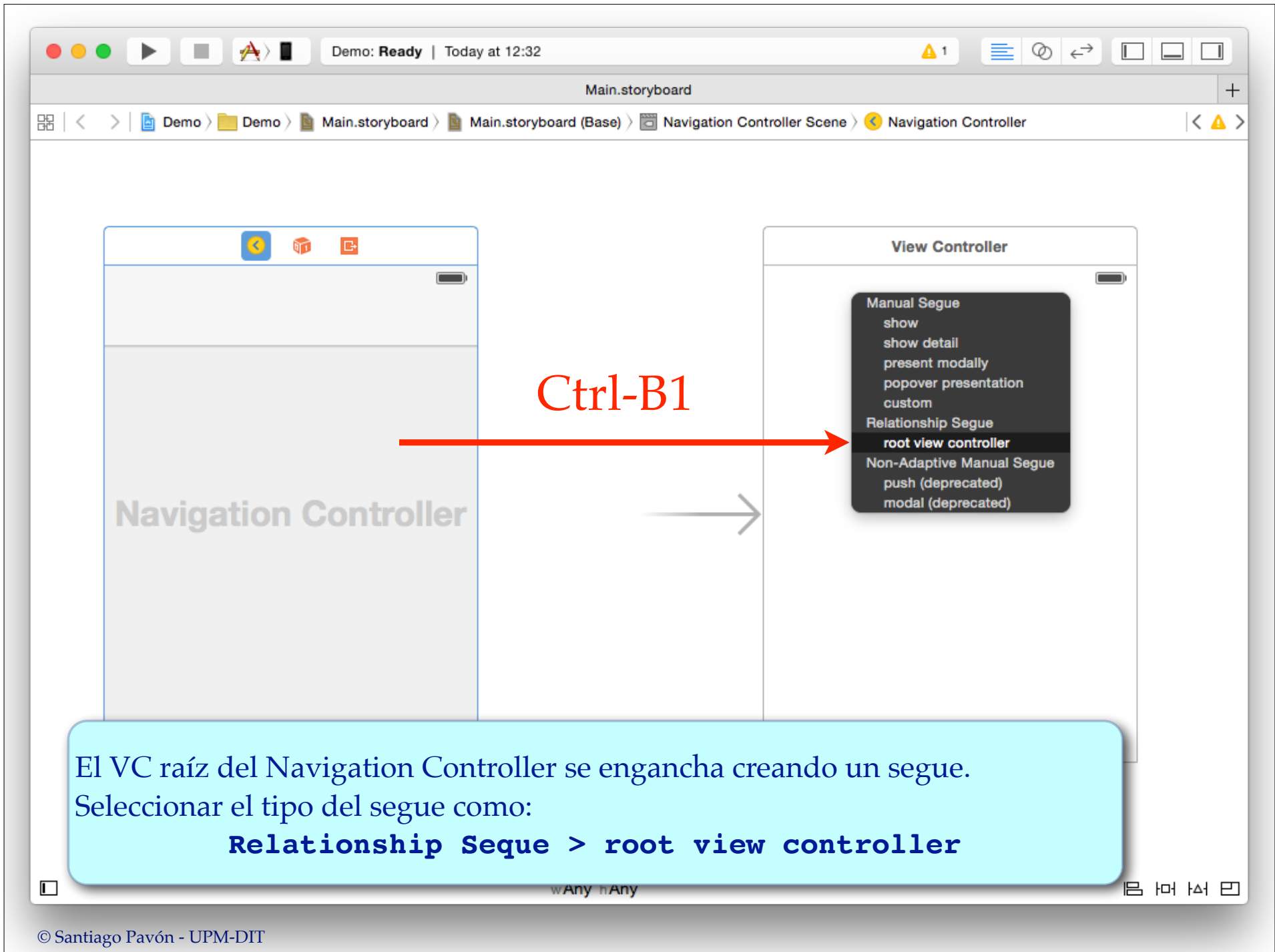
At the bottom of the sidebar, there are two informational cards:

- View Controller** - A controller that supports the fundamental view-management model in iOS.
- Navigation Controller** - A controller that manages navigation through a hierarchy of views.

A light blue callout box at the bottom of the screen contains the text: "Este es el resultado de embeber un VC en un Navigation Controller ejecutando: **Menú Editor > Embed In > Navigation Controller**".

Storyboard

- Editar los ficheros storyboard para:
 - Enganchar un VC ya existente a un Navigation Controller ya existente.
 - Creando un segue **Relationship** de tipo **root view controller** desde el Navigation Controller hasta el VC.
 - Enganchar entre sí los VC existentes para formar las cadenas de navegación.
 - Creando un segue de tipo Show para cada transición.
 - Origen: botón, celda de tabla, accesorio de la celda de una tabla, vc, reconocedor de gestos, ...
 - Destino: el VC destino.
 - Identificador: valor único que identifique el segue.



The image shows the Xcode storyboard editor interface. The main canvas displays a storyboard with two scenes: a Navigation Controller and a View Controller. A red curved arrow points from the View Controller scene to the Navigation Controller scene, indicating a transition. A blue callout box with a white background and a blue border contains the text "Cambiar la escena inicial al Navigation Controller". Another blue callout box with a white background and a blue border contains the text "Se pinta una barra de navegación automáticamente." The right sidebar shows the "View Controller" settings, with the "Is Initial View Controller" checkbox checked and circled in red. The "Transition Style" is set to "Cover Vertical". The bottom status bar shows "wAny hAny".

© Santiago Pavón - UPM-DIT

Storyboard

- Editar los ficheros storyboard para:
 - Enganchar un VC ya existente a un Navigation Controller ya existente.
 - Creando un segue de tipo **Relationship root view controller** desde el Navigation Controller hasta el VC.
 - Enganchar entre sí los VC existentes para formar las cadenas de navegación.
 - Creando un segue de tipo **Show** para cada transición.
 - **Origen:** botón, celda de tabla, accesorio de la celda de una tabla, vc, reconocedor de gestos, ...
 - **Destino:** el VC destino.
 - **Identificador:** valor único que identifique el segue.

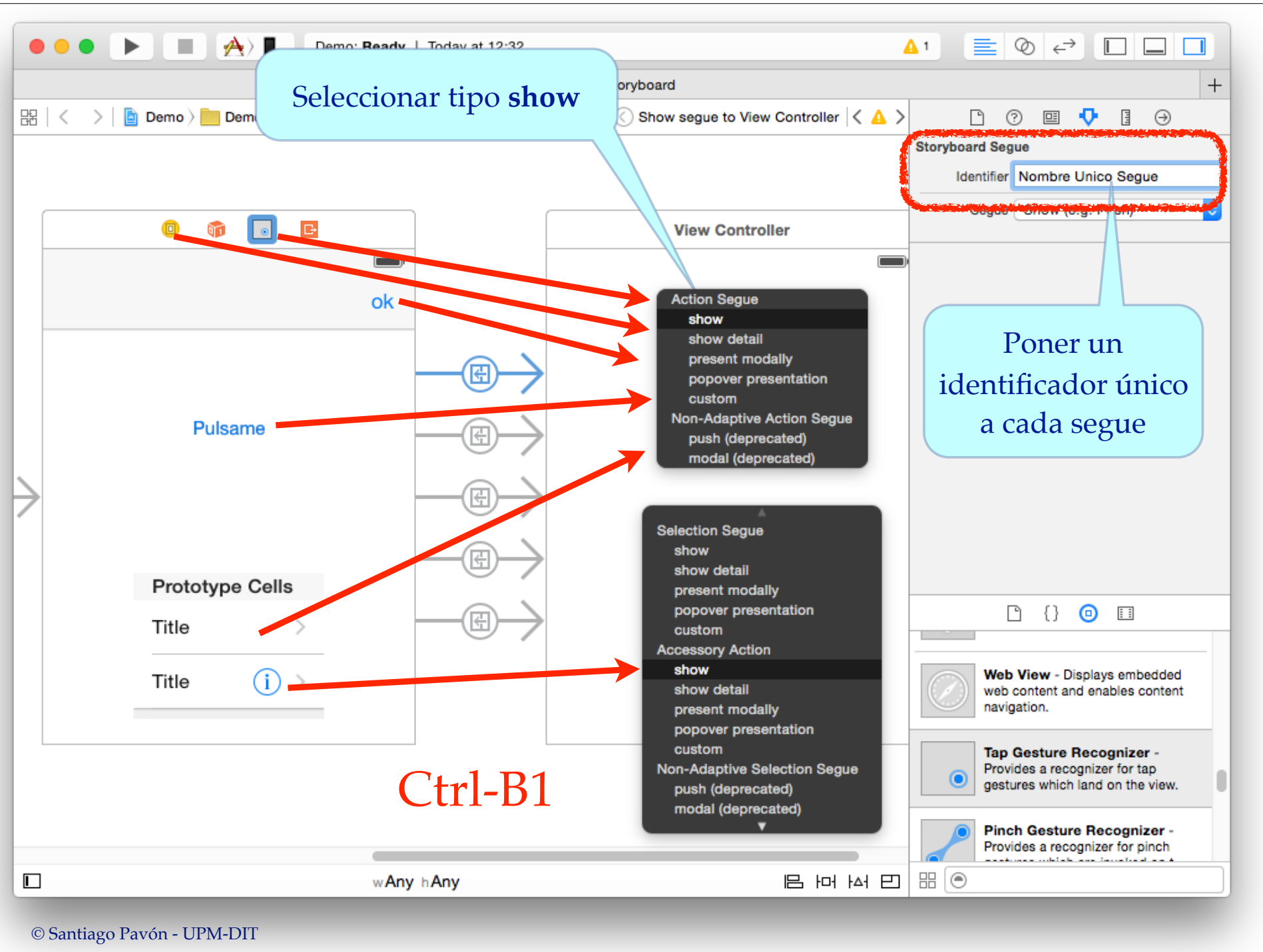
Seleccionar tipo show

Storyboard Segue
Identifier Nombre Unico Segue

Poner un
identificador único
a cada segue

Pulsame

Ctrl-B1



Configuraciones del Nav Controller

- Con IB podemos editar los ficheros storyboard para configurar los Navigation Controller y los VC que presenta:
 - Añadir objetos Navigation Item a los VC que lo necesiten.
 - Poner un título en la barra de navegación de cada VC.
 - Estilo de la Navigation Bar.
 - Visibilidad de la Navigation Bar y de la Tool Bar.
 - Ocultar las barras con un tap, swipe, al mostrar el teclado, o al pasar a un Size Class vertical compacto.
 - ...

Running Demo on iPhone 6

Main.storyboard

Demo > Demo > Main....board > Main....Base > Navigation Controller Scene > Navigation Controller

Navigation Controller

UNO

DOS

Navigation Controller

UNO

DOS

Navigation Controller

Bar Visibility Shows Navigation Bar
 Shows Toolbar

Hide Bars On Swipe
 On Tap
 When Keyboard Appears
 When Vertically Compact

View Controller

Title

Is Initial View Controller

Layout Adjust Scroll View Insets

mechanism for displaying a navigation bar just below the sta...

Navigation Item - Represents a state of the navigation bar, including a title.

Toolbar - Provides a mechanism for displaying a toolbar at the bottom of the screen.

Bar Button Item - Represents an

wAny hAny

Demo

UINavigationController

- Propiedades de **UINavigationController** :

- **topViewController** apunta al VC en el top de la pila.
- **visibleViewController** apunta al VC mostrado.
- **viewControllers** es el array con la pila de VCs.
- **navigationBar** es la barra de navegación mostrada.
- **toolbar** es la barra de herramientas mostrada.
- etc...

- Métodos de **UINavigationController** usados para navegar:

```
func popViewControllerAnimated(_ animated: Bool)  
    -> UINavigationController?
```

```
func popToRootViewControllerAnimated(_ animated: Bool)  
    -> [AnyObject]?
```

```
func pushViewController(_ viewController: UINavigationController,  
    animated animated: Bool)
```

etc...

UINavigationController

- Los objetos **UINavigationController** gestionan una pila con los objetos UINavigationControllerItem por los que se va navegando.
 - Cada objeto UINavigationControllerItem apunta al VC que debe mostrar cuando es el primer elemento de la pila.
- Es usado por los navigation controllers para implementar su interface de navegación.
 - pero las barras de navegación también pueden usarse de forma independiente.

UINavigationController

- Los objetos UINavigationController contienen los botones y views que hay que mostrar en una UINavigationController.
- Cada objeto View Controller usado en este tipo de navegación tiene asociado un objeto UINavigationController donde configura los elementos a mostrar en la barra de navegación cuando es visible.
 - **title** es el título mostrado en el centro de la barra de navegación.
 - **backBarButtonItem** es el botón para retroceder a la pantalla anterior.
 - **prompt** es una línea de texto situada en la parte superior de la barra de navegación
 - **titleView**, **leftBarButtonItem**, **leftBarButtonItem**, **rightBarButtonItem**, **rightBarButtonItem** son views y botones personalizados.
 - ...

UIToolbar

- **UIToolbar** es una barra de herramientas.
 - Por herramientas queremos decir botones **UIBarButtonItem**.
- Los **UINavigationController** muestran en una barra de herramientas los elementos contenidos en la propiedad **toolbarItems** del VC actual.
 - Cada VC tiene su propia propiedad **toolbarItems**.
 - **toolbarItems** es un array de objetos **UIBarButtonItem**.
 - Los **UIBarButtonItem** pueden ser:
 - botones que usan target/ action o segues:
 - estilos: con borde, planos, con texto, con imágenes predefinidas, texto personalizado o predefinido, etc.
 - botones usados para crear separaciones fijas o flexibles.
 - La visibilidad de esta barra se controla con la propiedad **toolbarHidden**.
- También podemos añadir nuestras propias **UIToolbars** a nuestras GUIs.
 - Típicamente en la parte superior o inferior de la pantalla.

UIViewController

- Los objetos **ViewControllers** tienen:
 - Propiedad **navigationController** que apunta al **UINavigationController** que los controla.
 - Y la propiedad **parentViewController** también apunta al **Navigation Controller**.
 - Propiedad **title** usada en la barra de navegación para poner el título.
 - Propiedad **toolbarItems** es un array con los items a poner en la toolbar.
 - Propiedad **navigationItem** gestiona los botones y vistas que se muestran en la barra de navegación.
 - ...

Ejemplo: Poner un Botón

- Con Interface Builder:
 - Añadir un botón a la barra de navegación:
 - Editar storyboard con Interface Builder.
 - Arrastrar un UIBarButtonItem desde la biblioteca de objetos a la barra de navegación.
 - Configurar propiedades del botón.
 - Crear IBActions o segues

- Programáticamente:

```
let b = UIBarButtonItem(title: "acercar",
                       style: .Bordered,
                       target: self,
                       action: "acercar:")
vc.navigationItem.leftBarButtonItem = b
```

- Existen más métodos inicializadores que permiten especificar:
 - imágenes, símbolos del sistema, views personalizadas.

Flujo de Datos

- Normalmente, los view controllers por los que navego necesitan pasarse datos entre ellos.
 - Desde el view controller actual quiero parametrizar el siguiente view controller a mostrar.
- Buen diseño: poca cohesión entre los VC:
 - Evitar efectos laterales, facilitar la reutilización, VC independientes, ...
 - Evitar usar variables globales, objetos singletons, el delegado de la aplicación, etc. para pasar datos entre los VC.
- Para pasar datos al siguiente view controller:
 - ej: usar parámetros de entrada, protocolo data source.
- Para devolver datos al view controller anterior.
 - ej: usar delegación, unwind segues, closures.

