



POLITÉCNICA

ETSIT  
UPM

*dit*  
UPM

# Desarrollo de Apps para iOS

## Entrada de Texto

IWEB 2016-2017  
Santiago Pavón

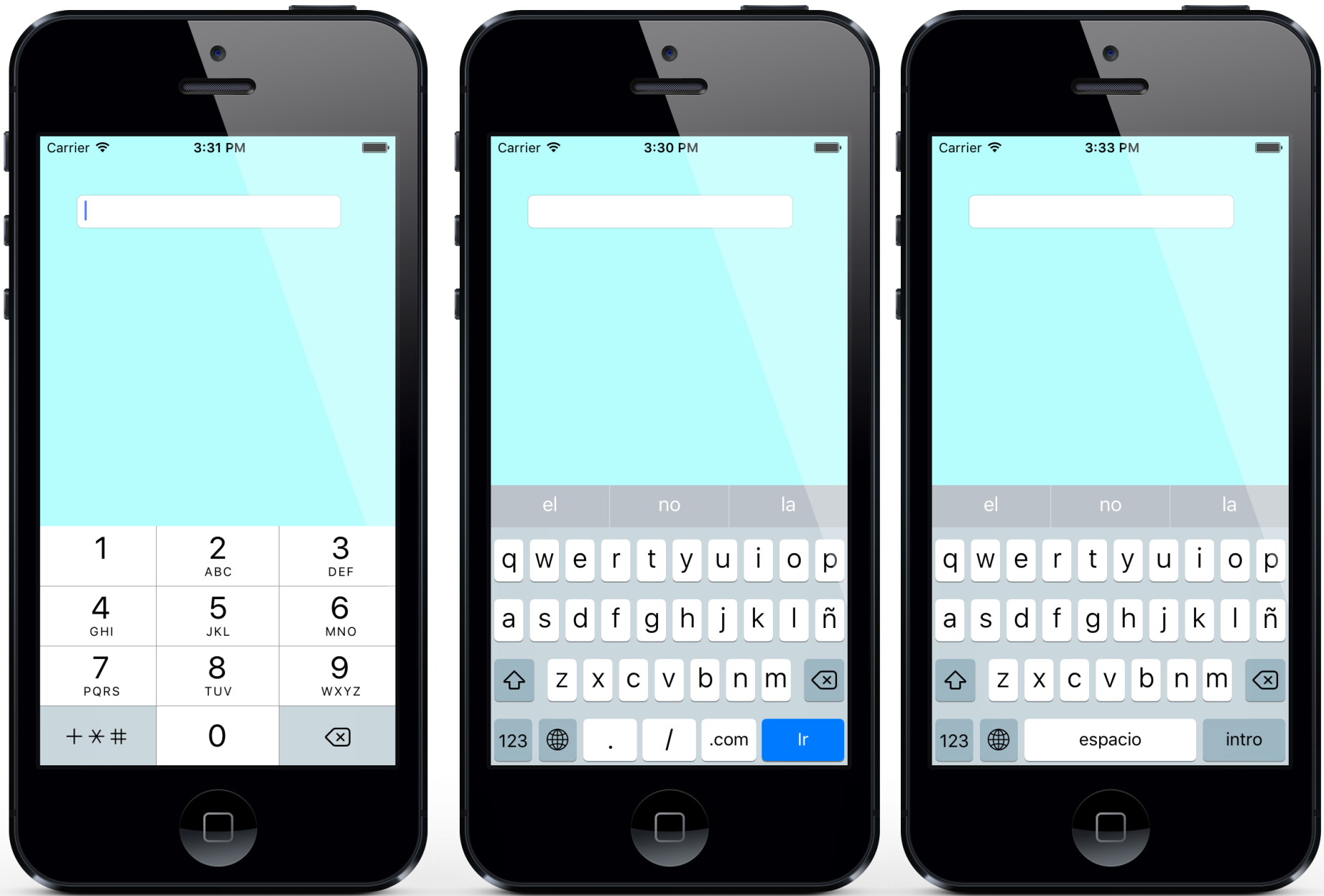
ver: 2016.09.13

# UITextField y UITextView

- Las views de entrada de texto son:
  - **UITextField** - editar una línea de texto.
  - **UITextView** - editar múltiples líneas, permite scroll.
- Cuando una view de entrada texto se convierte en el **primer responder**, el teclado aparece automáticamente.
  - La view se convierte en el primer responder cuando:
    - El usuario toca con el dedo sobre ella
    - o se ejecuta su método **becomeFirstResponder**.
- El teclado desaparece cuando la view deja de ser el primer responder.
  - Tocando sobre otra view para que se convierta en el primer responder
  - o ejecutando su método **resignFirstResponder**.

# El Teclado

- El teclado ya está hecho, pero podemos configurar muchas características del teclado asignando valores a las propiedades definidas por el protocolo **UITextInputTraits**.
  - ascii, numérico, mail, url, tipo de tecla return, capitalización automática, corrección del texto, seguridad para passwords, etc.
  - Estas propiedades también pueden configurarse desde el interface builder.
  - Este protocolo lo implementan **UITextField** y **UITextView**.
- El teclado también puede ampliarse con controles creados por el usuario.
  - Views accesorias añadidas usando la propiedad **inputAccessoryView** de UITextField o UITextView.
    - Se muestran encima del teclado.
- También podemos crearnos nuestro propio teclado asignándolo a la propiedad **inputView** de UITextField o UITextView.



# Delegados

- Los controles de entrada de texto pueden tener delegados a los que informar sobre las actividades relacionadas con la edición.
- Los protocolos delegados son: **UITextFieldDelegate** y **UITextViewDelegate**.
- Estos protocolos definen métodos para:
  - Consultar si se permite que comience la edición de un campo
    - func textFieldShouldBeginEditing(:)**
    - func textViewShouldBeginEditing(:)**
  - Informar de que ha empezado la edición de un campo de texto
    - func textFieldDidBeginEditing(:)**
    - func textViewDidBeginEditing(:)**
  - Consultar si se permite que termine la edición de un campo
    - textFieldShouldEndEditing(:)**
    - textViewShouldEndEditing(:)**
  - Informar de que ha terminado la edición de un campo de texto
    - Este suele ser el método utilizado para acceder a la propiedad **text** y extraer el texto introducido.
    - textFieldDidEndEditing(:)**
    - textViewDidEndEditing(:)**
  - Consultar al delegado si debe atenderse a la pulsación de la tecla **Return**.
    - textFieldShouldReturn(:)**
  - ...



# Más Propiedades / Métodos

- **UITextField** tiene propiedades para:
  - indicar si se debe borrar el texto existente al comenzar una nueva edición  
**clearsOnBeginEditing**
  - Indicar el tamaño del texto  
**adjustsFontSizeToFitWidth, minimumFontSize**
  - Texto a mostrar cuando el campo está vacío  
**placeholder**
  - Imágenes usadas para la apariencia  
**background, disabledBackground**
  - Overlay views  
**clearButtonMode, leftView, leftViewMode, rightView, rightViewMode**
  - ...
- **UITextView** tiene un método para hacer scroll a un rango del texto  
**scrollRangeToVisible(:)**
- Y ambos tienen también propiedades para indicar colores, fonts, alineamiento de texto, etc.

# Notificaciones Teclado

- Para conocer el estado del teclado podemos escuchar las notificaciones que envía el objeto `view.window` (es el `UIWindow` de la aplicación):

`UIKeyboardWillShow`

`UIKeyboardDidShow`

`UIKeyboardWillHide`

`UIKeyboardDidHide`

• • •

- Suelen usarse para reposicionar los elementos que tapa el teclado al aparecer, o recolocarlos en su posición inicial cuando éste desaparece.

# Demo

- Crear un objeto **UITextField** para editar una línea de texto
  - crear un outlet que lo apunte y llamarlo **textField**
- Crear un objeto **UIButton**.
  - La etiqueta del botón será "done", ya que al pulsarlo terminará la edición, y se ocultará el teclado
  - Esto se consigue conectando el botón con una acción que ejecute:  
**textField.resignFirstResponder()**
    - No poner argumentos a la acción creada para poder reutilizarla en los puntos siguientes.
- Conectar también el evento **Did End On Exit** del **UITextView** con la acción anterior.
  - Al pulsar **Return** (o similar) se llamará a la misma acción y se ocultará el teclado
- Convertir el **UIView** del fondo en un **UIControl**.
  - Conectar su evento **Touch Down** con la acción anterior.
    - Así al tocar el fondo de la pantalla también se ocultará el teclado.
- Jugar con las características (traits) del teclado.
- ...



