



# Tema 2

## Interfaces gráficas

# Interfaces gráficas



# Interfaces gráficas

- Una vez explicados los patrones y los view controllers podemos pasar a la creación de interfaces.
- Para poder llevar a cabo esto, debemos conocer los archivos que usaremos y ciertas características que poseen así como la herramienta que usaremos para crearlas.

# Archivos XIB



# Archivos XIB

- Los archivos XIB contienen la interfaz gráfica de la aplicación. Corresponden con la vista en el patrón MVC.
- Son archivos XMLs por lo que pueden generarse a mano.
- Puesto que la creación a mano es muy costosa, existe una herramienta llamada Interface Builder que permite modificarlos simplemente arrastrando los elementos que queremos que compongan una vista.

# Archivos XIB

- Al ser archivos XML puede ser editados con cualquier editor de texto.

```
<?xml version="1.0" encoding="UTF-8"?>
<archive type="com.apple.InterfaceBuilder3.CocoaTouch.XIB" version="7.10">
  <data>
    <int key="IBDocument.SystemTarget">784</int>
    <string key="IBDocument.SystemVersion">10A405</string>
    <string key="IBDocument.InterfaceBuilderVersion">732</string>
    <string key="IBDocument.AppKitVersion">1031</string>
    <string key="IBDocument.HIToolboxVersion">432.00</string>
    <object class="NSMutableDictionary" key="IBDocument.PluginVersions">
      <string key="NS.key.0">com.apple.InterfaceBuilder.IBCocoaTouchPlugin</string>
      <string key="NS.object.0">62</string>
    </object>
    ...
  </data>
</archive>
```

# Outlets y Actions



# Outlets y Actions

- Los outlets son un mecanismo utilizado para referenciar elementos pertenecientes a las vistas.
- Las actions son otro mecanismo que permite invocar ciertos métodos cuando se capturan eventos en la interfaz como cuando el usuario pulsa un botón.



# Outlets

- Los outlets se declaran mediante la palabra clave IBOutlet

```
<?xml version="1.0" encoding="UTF-8"?>
<archive type="com.apple.InterfaceBuilder3.CocoaTouch.XIB" version="7.10">
  <data>
    <int key="IBDocument.SystemTarget">784</int>
    <string key="IBDocument.SystemVersion">10A405</string>
    <string key="IBDocument.InterfaceBuilderVersion">732</string>
    <string key="IBDocument.AppKitVersion">1031</string>
    <string key="IBDocument.HIToolboxVersion">432.00</string>
    <object class="NSMutableDictionary" key="IBDocument.PluginVersions">
      <string key="NS.key.0">com.apple.InterfaceBuilder.IBCocoaTouchPlugin</string>
      <string key="NS.object.0">62</string>
    </object>
    ...
  </data>
</archive>
```

# Outlets

- Declarar una variable con IBOutlet no afecta a su contenido o comportamiento.
- La palabra clave IBOutlet simplemente se utiliza para avisar al compilador y a la vista de que el objeto puede ser conectado a un elemento de la interfaz incluido en un archivo XIB.

# Actions

- Las actions se declaran mediante la palabra clave `IBAction` donde debería de ir el valor de retorno de un método.

```
-(IBAction)myButtonPressed:(id)sender;
```

# Actions

- Al igual que con los IBOutlets, declarar un método como IBAction no afecta a su comportamiento, únicamente se emplea para indicar al editor de interfaces que es un método que puede enlazarse con un elemento al recibir ciertos tipos de eventos.

# Interface Builder

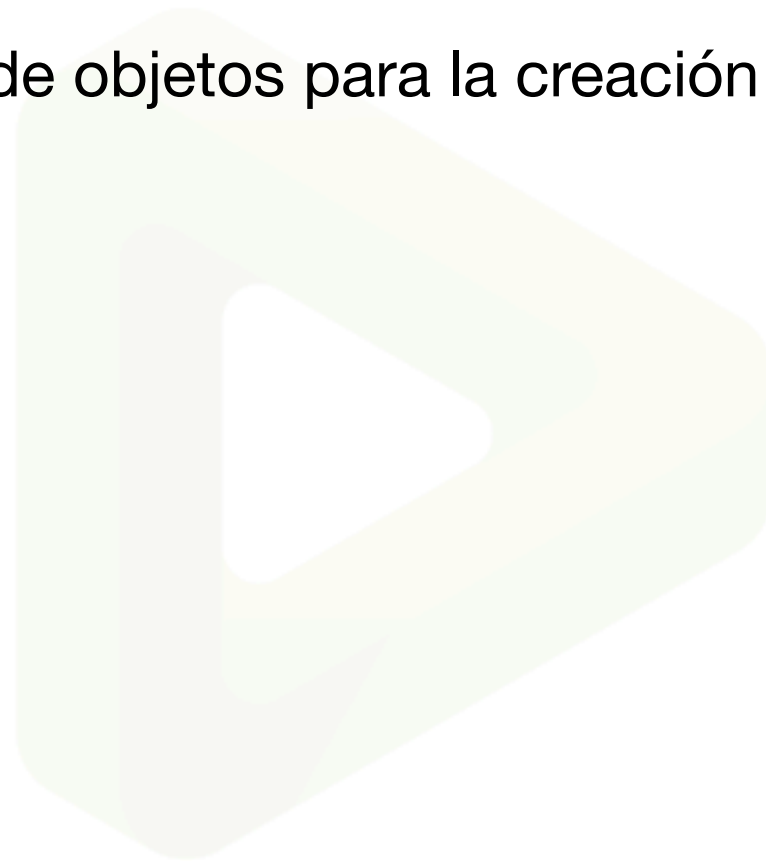


# Interface Builder

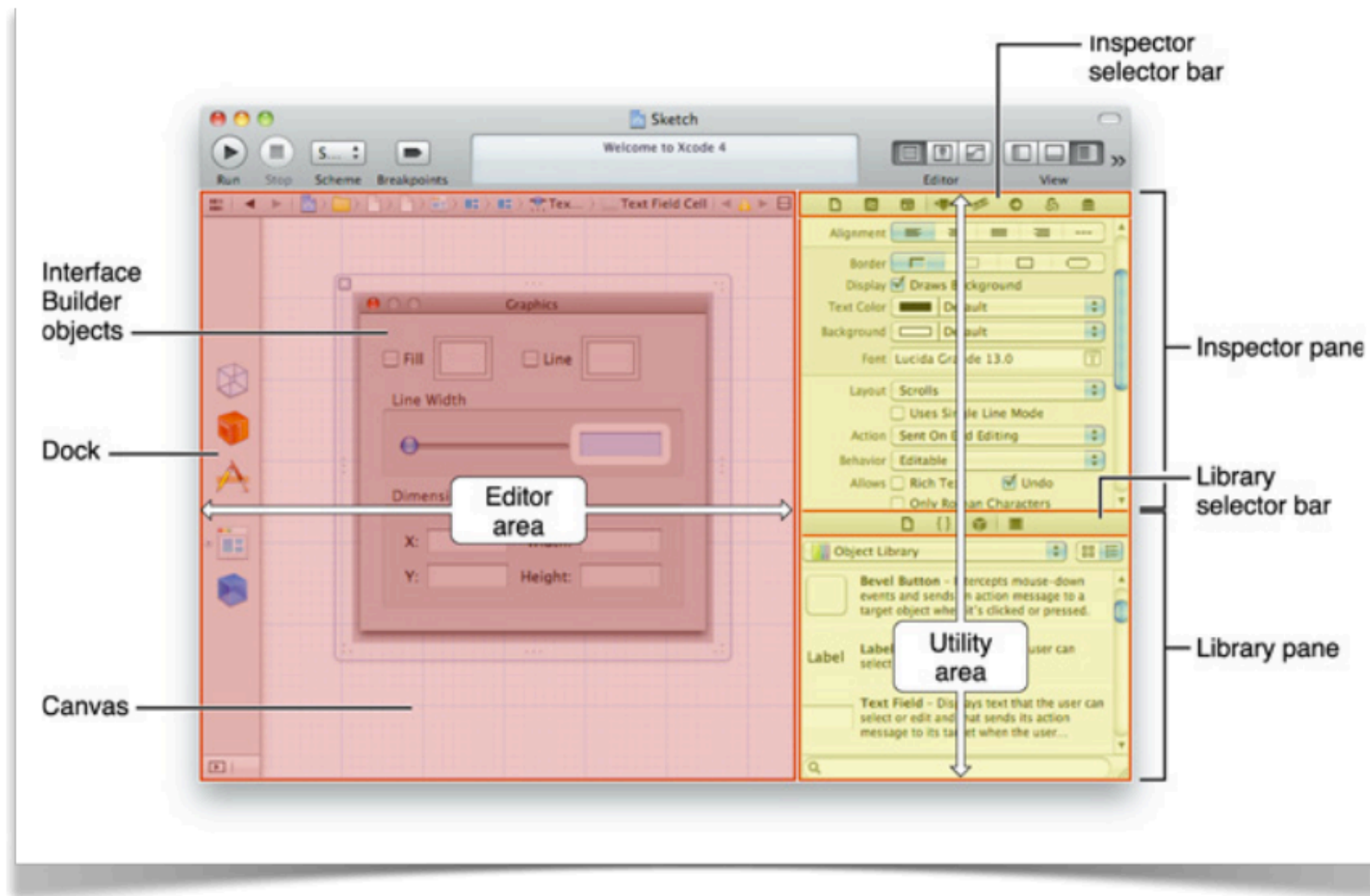
- Interface Builder es el editor gráfico de las interfaces de la aplicación.
- La edición de las interfaces se realiza en los archivos XIB.
- Interface Builder está integrado dentro de Xcode por lo que simplemente seleccionando un archivo XIB se abrirá.

# Interface Builder

- Provee colecciones de objetos para la creación de interfaces.



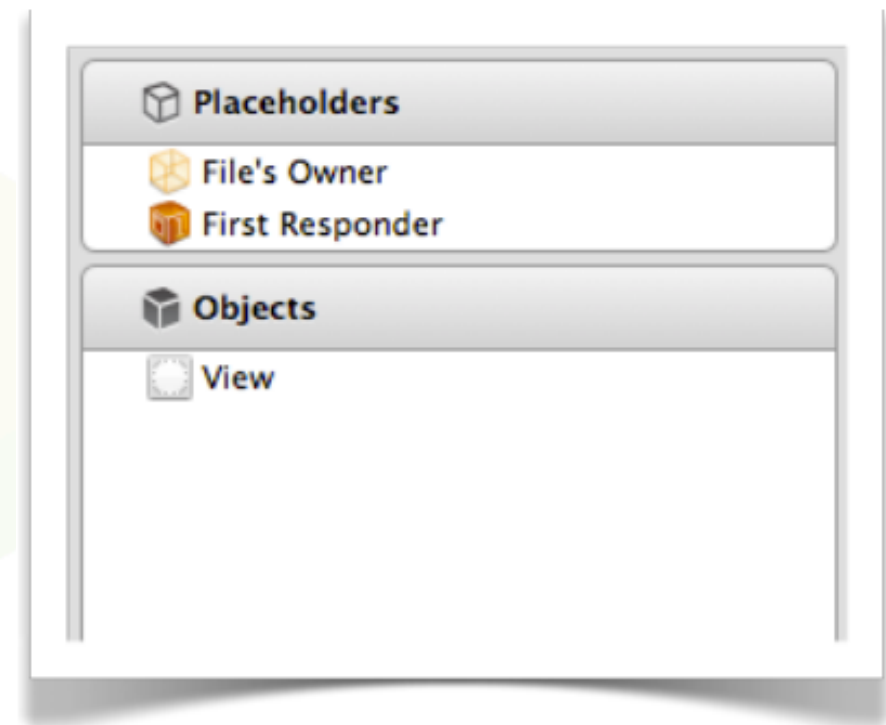
# Interface Builder





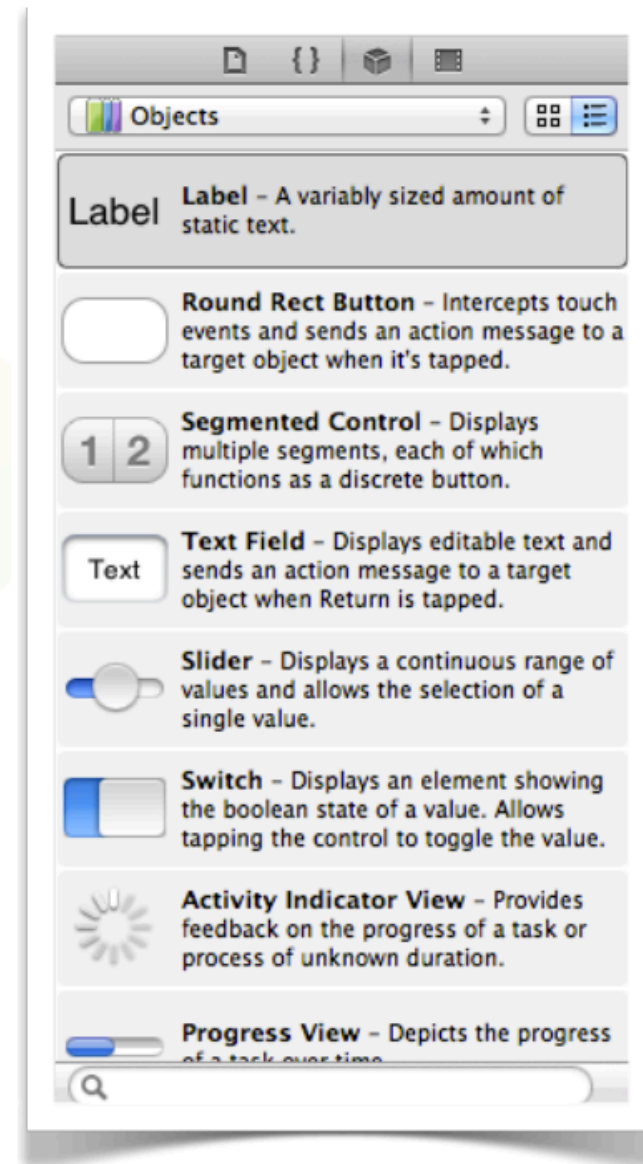
# Interface Builder - Dock

- El elemento File's Owner representa el objeto asignado al propietario de la interfaz gráfica.
- First Responder representa el objeto con el que el usuario interacciona.
- View es el objeto base de las interfaces. Contiene el resto de elementos que forman parte de la interfaz gráfica.



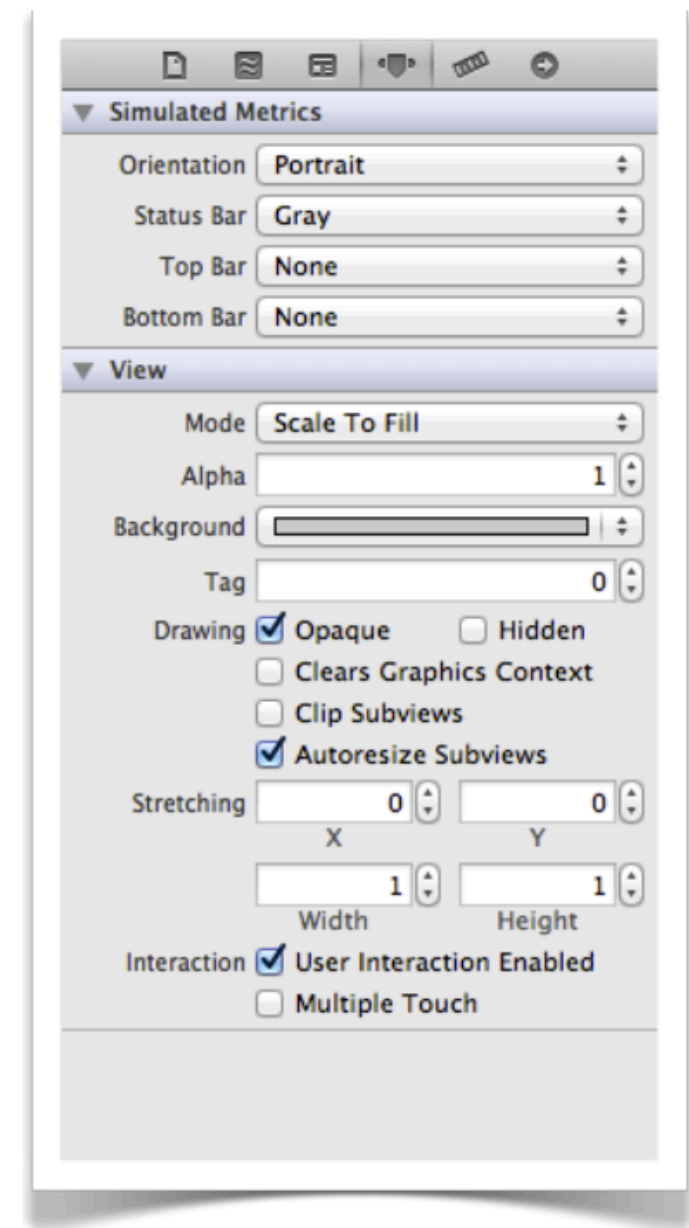
# Interface Builder - Library Panel

- Contiene los objetos y elementos que se pueden añadir a la interfaz gráfica.
- Es sensible al contexto por lo que según el documento abierto se mostrarán unos componentes u otros.
- Basta con arrastrar el elemento a la vista para añadir nuevos componentes.
- Permite buscar objetos en la librería.



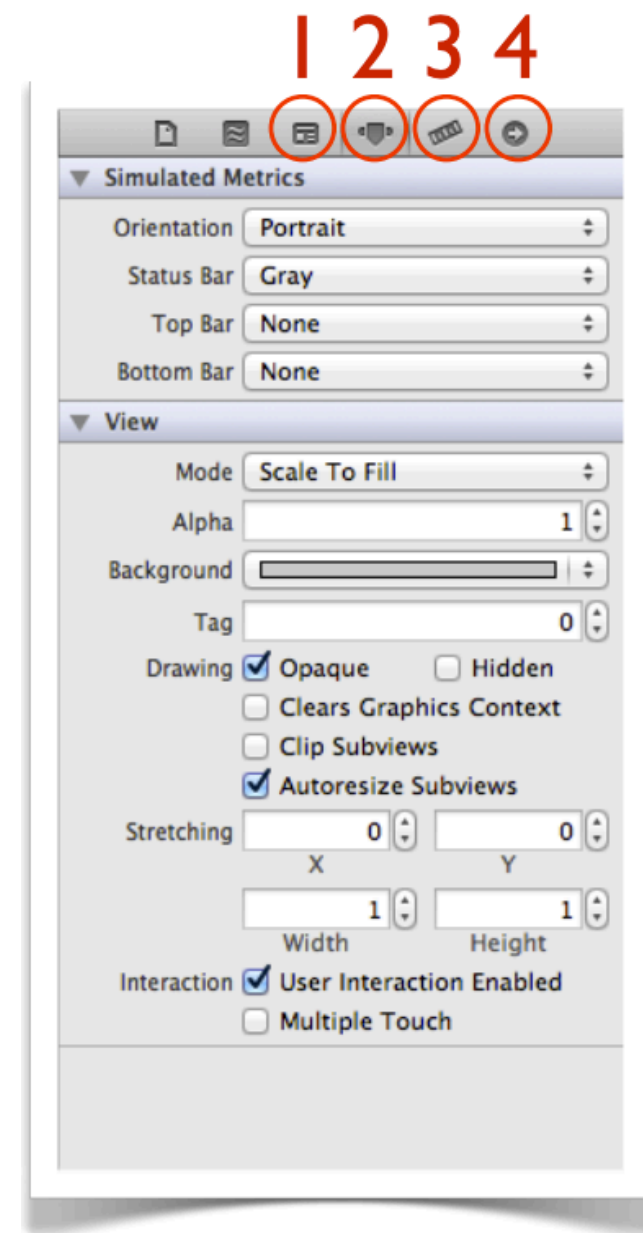
# Interface Builder - Inspector Panel

- Esta zona del Interface Builder permite visualizar los atributos del componente seleccionado.
- Esta dividida en pestañas que agrupan distintas propiedades y ajustes del componente.



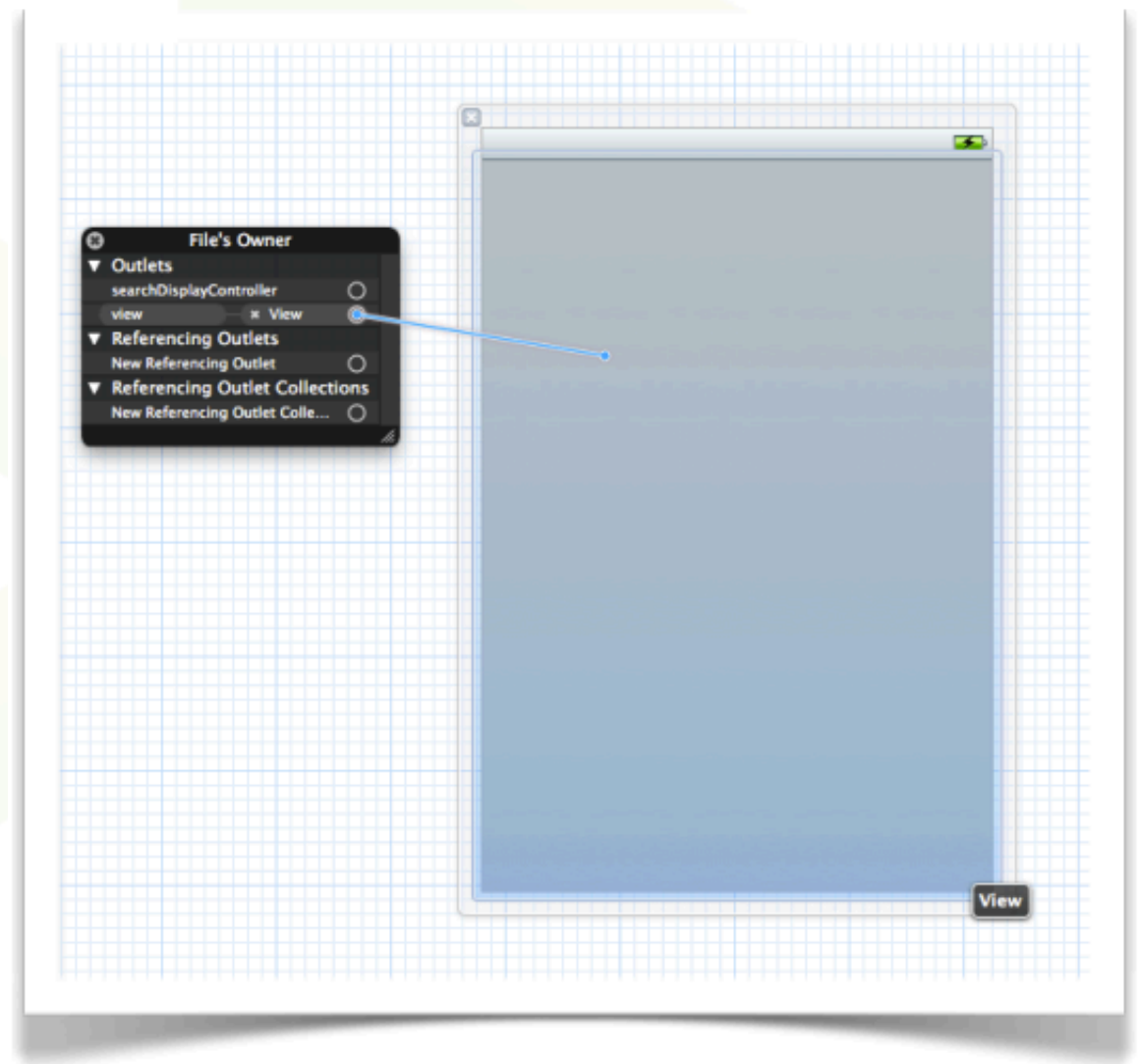
# Interface Builder - Inspector Panel

1. Sección que identifica el objeto
2. Configuración de los atributos del objeto
3. Tamaño, posición y ajustes de escalado del objeto
4. IBOutlets y IBActions enlazados



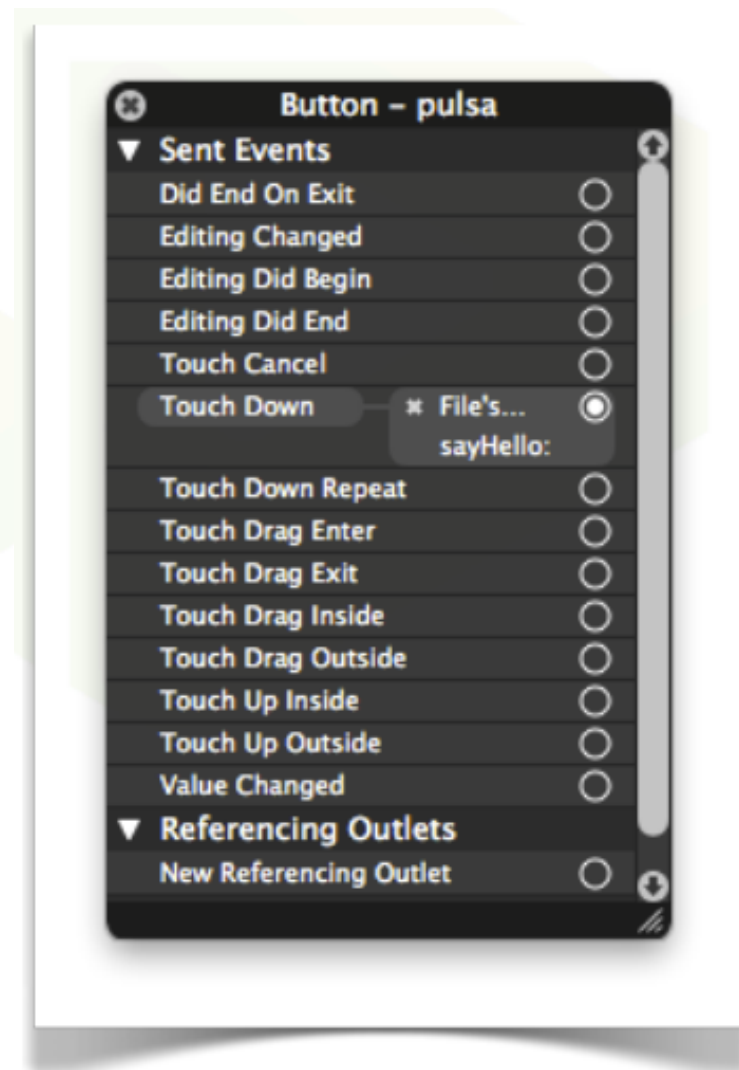
# Interface Builder - Conexiones

- Las conexiones enlazan los IBOutlet declarados en los controladores con los objetos de la interfaz gráfica que los representa



# Interface Builder - Conexiones

- También es posible enlazar los elementos con las IBActions correspondientes.



# Storyboards



# Storyboards

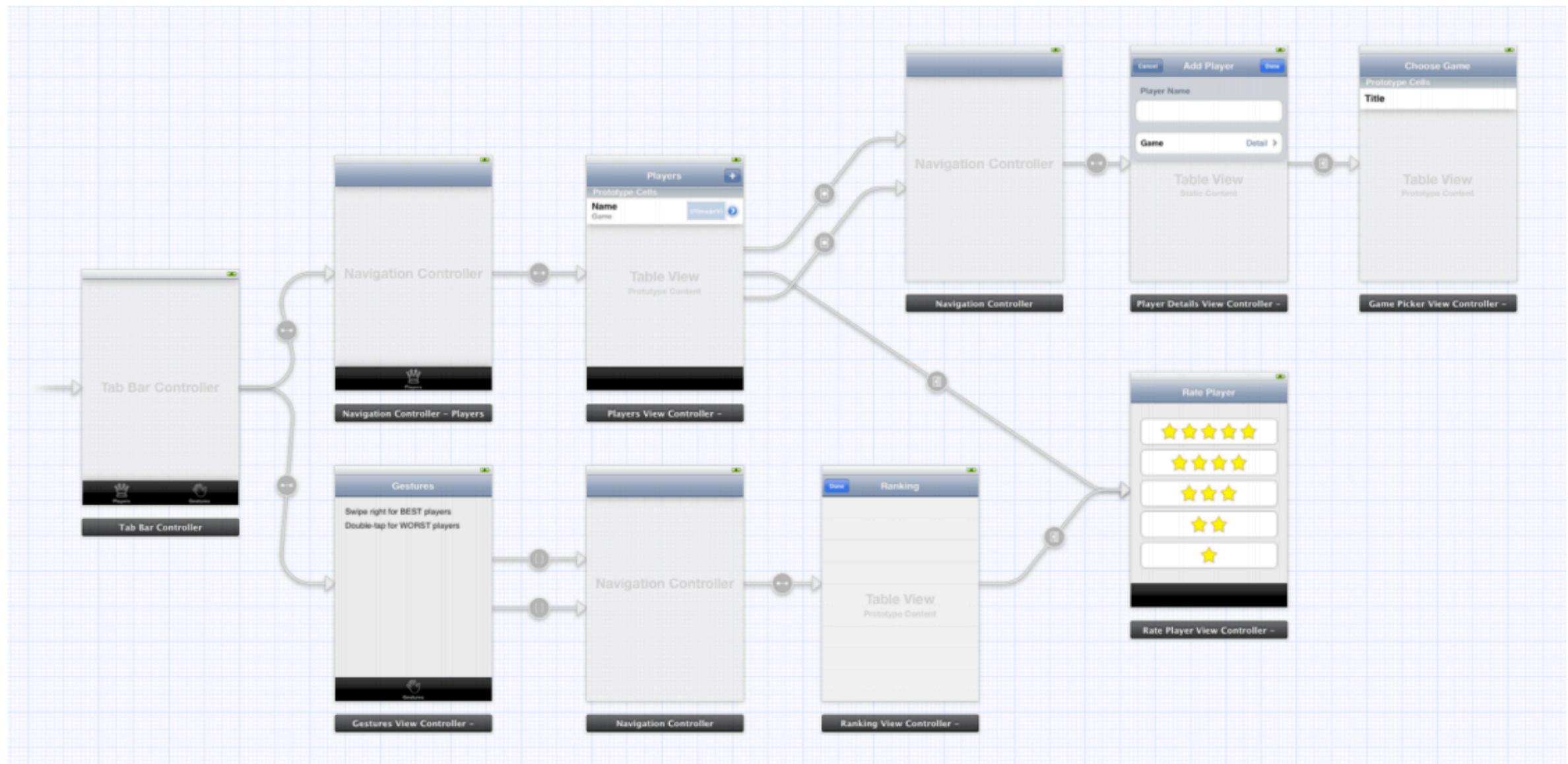
- Un storyboard es un fichero que agrupa todas las vistas de un proyecto.
- A diferencia de un archivo XIB que solo representa la vista de un controlador, en el storyboard se crean todas las vistas necesarias.
- Para modificarlos se usa Interface Builder por lo que todas las características explicadas pueden aplicarse a los storyboards.



# Storyboards

- Además de permitir crear las diferentes vistas, es posible crear las relaciones y transiciones entre las distintas vistas usadas en la aplicación.
- Mediante su uso, reducimos el número de archivos de interfaz gráfica que debemos crear, así como permite una fácil visualización de toda la estructura del proyecto y su navegación entre vistas.
- Las relaciones y transiciones entre vistas se crean mediante "segues".

# Storyboards



# Storyboards

- Con el uso de storyboards debemos tener en cuenta que es en este fichero donde se indica el controlador inicial.
- Por defecto viene marcado mediante una flecha sin inicio que apunta a uno de los controladores. Puede modificarse el controlador inicial mediante la pestaña de atributos en el Inspector Panel.

