

Tutorial Móvil: Creando una Aplicación FireMonkey Mobile (iOS and Android)

Este tópico describe cómo crear una aplicación móvil FireMonkey "Hello World" ya sea para la plataforma de destino iOS o Android.

Antes de empezar

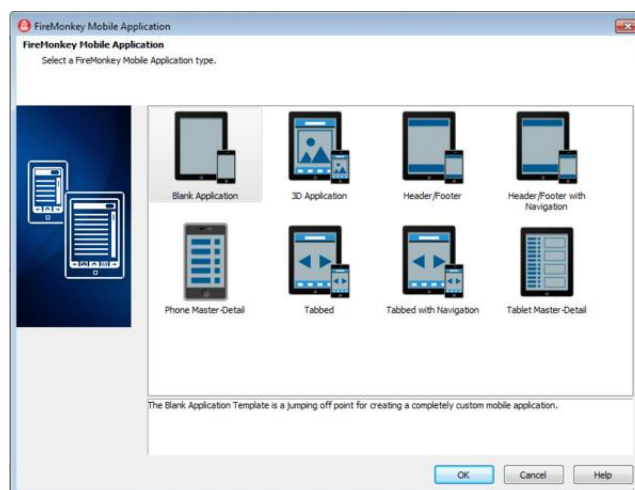
Para desarrollar aplicaciones móviles (iOS y Android) usando RAD Studio, usted necesita completar algunos pasos importantes de configuración. Este tutorial asume que usted ya ha completado todos los pasos de configuración necesarios.

Por detalles, vea:

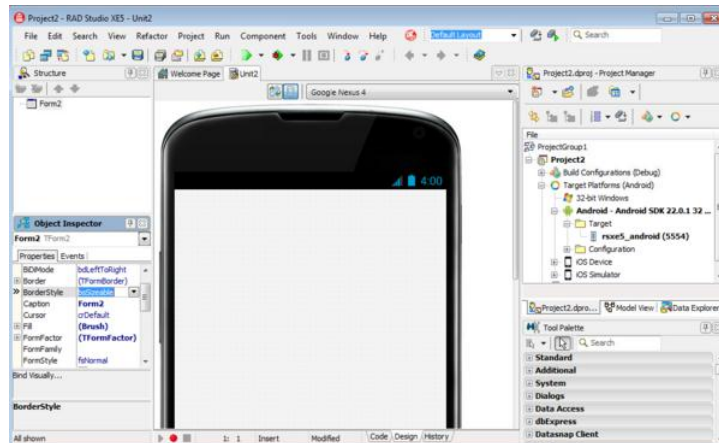
- [Tutorial Móvil: Configurando Su Entorno de Desarrollo sobre la Mac \(iOS\)](#)
- [Tutorial Móvil: Configurando Su Entorno de Desarrollo sobre Windows PC \(iOS\)](#)
- [Tutorial Móvil: Configurando Su Entorno de Desarrollo sobre Windows PC \(Android\)](#)

Paso 1: Crear una Nueva Aplicación FireMonkey para Android o iOS

1. Seleccione **File > New > FireMonkey Mobile Application - Delphi:**



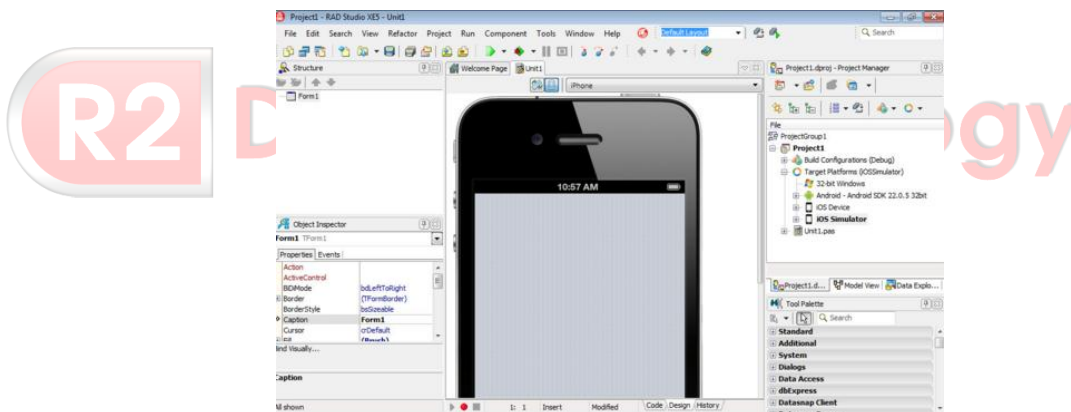
2. Seleccione **Blank Application**. El **FireMonkey Mobile Form Designer** muestra un nuevo formulario para la plataforma móvil por defecto, **Android**:



3. **Android:** Si desea crear una app **Android** que se ejecute por defecto sobre el emulador Android, puede usar el preseleccionado target platform.

Para información acerca de usar un dispositivo Android, vea [Configurando Su Sistema para Detectar Su Dispositivo Android](#).

4. **iOS:** Si desea crear una app **iOS**, abrir el nodo **Target Platform** en el **Project Manager** y doble-click en **iOS Simulator**:

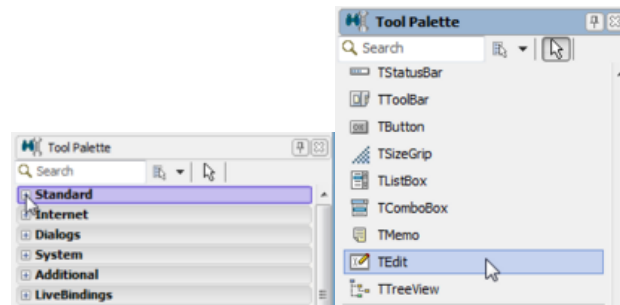


Paso 2: Colocar Componentes sobre el Formulario Móvil FireMonkey

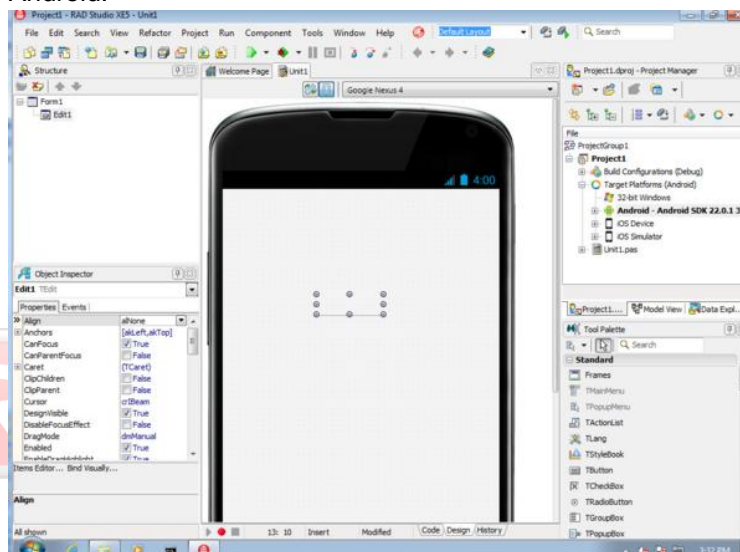
El primer paso en crear una aplicación FireMonkey Mobile es diseñar la interface de usuario; estes es también el primer paso cuando usted apunta a plataformas de escritorio. Hay muchos componentes reusables disponibles en el IDE para crear interfaces de usuario.

1. Mover el puntero del ratón sobre la **Tool Palette**, y expanda la categoría **Standard** haciendo click en el ícono mas (+) al lado del nombre de la categoría.

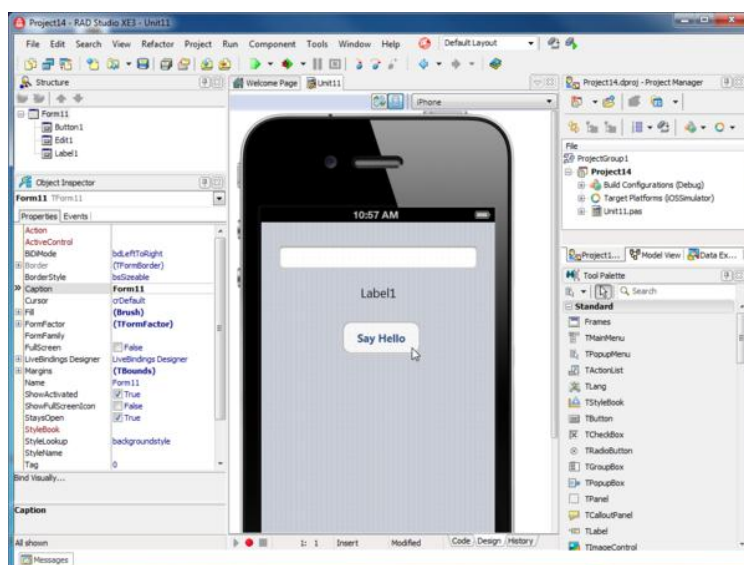
2. Seleccione el componente **TEdit** y haga doble-click sobre el TEdit o suéltelo sobre el FireMonkey Mobile Form Designer.



Una instancia del componente TEdit aparece sobre el formulario. Aquí está una app Android:



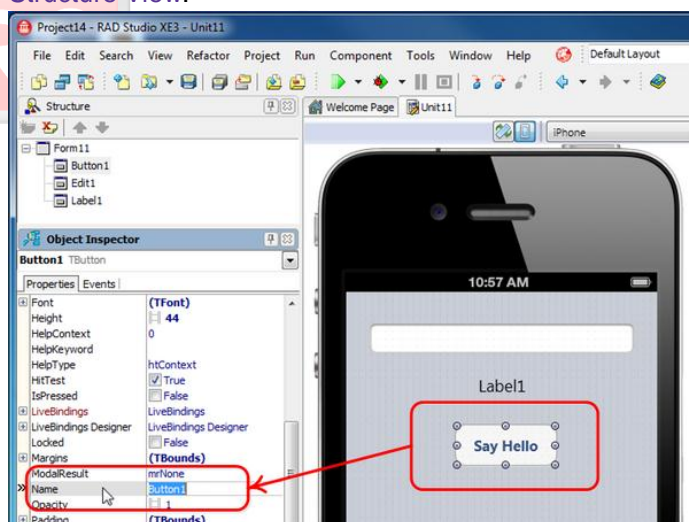
3. Repetir estos pasos, pero ahora agregue un componente **TLabel** y un **TButton** al formulario.
4. Seleccione el botón y cambié la propiedad **Text** en el **Object Inspector** a "Say Hello".
5. Ahora debería ver tres componentes sobre el **FireMonkey Mobile Form Designer**. Aquí está la app iOS:



- Después que ha colocado esos componentes sobre el **FireMonkey Mobile Form Designer**, el IDE automáticamente coloca nombres para los componentes.

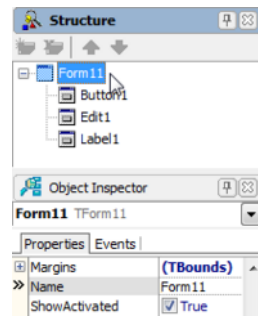
Para ver o cambiar el nombre a un componente, seleccione el componente sobre el Form Designer, y luego encuentre su propiedad **Name** en el **Object Inspector** y el

Structure View:



Para un componente **TButton**, el nombre del componente es colocado por defecto a **Button1** (o Button2, Button3, dependiendo de cuántos TButtons haya creado en esta aplicación).

- El formulario sobre el que ha colocado estos componentes también tiene un nombre. Seleccione el fondo del **FireMonkey Mobile Form Designer**, y seleccione la propiedad **Name** en el **Object Inspector**. El nombre del formulario **Form1** (o Form2, Form3,...) es mostrado. También puede localizar el nombre del formulario en el **Structure View**:



8. Puede cambiar fácilmente a source code (código fuente) seleccionando la pestaña **Code** en la parte inferior del Form Designer o presionando la tecla **F12**.

Puede cambiar entre el [FireMonkey Mobile Form Designer](#) y el [Code Editor](#) en cualquier momento:



Cuando cambia hacia el **Code Editor**, puede ver el código fuente que el IDE tiene generado. Puede encontrar tres componentes definidos (Edit1, Label1, y Button1):

```

Welcome Page | Unit11
- unit Unit11;
- interface
- uses
  System.SysUtils, System.Types, System.UITypes,
  System.Classes, System.Variants, FMX.Types,
  FMX.Controls, FMX.Forms, FMX.Dialogs, FMX.StdCtrls, FMX.Edit;
10 type
  TForm1 = class(TForm)
    Button1: TButton;
    Edit1: TEdit;
    Label1: TLabel;
    private
      { Private declarations }
    public
      { Public declarations }
    end;
20 var
  Form1: TForm1;
- implementation
  {$R *.fmx}
28 end.

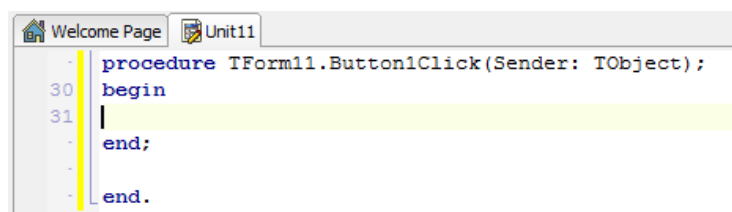
```

Note que la cláusula **Uses** es actualizado (para agregar la unidad FMX.StdCtrls para TLabel y la unidad FMX.Edit para TEdit) cuando usted guarde o ejecute su proyecto.

Paso 3: Escriba un Manejador de Evento en Delphi para un Click de Botón de parte del Usuario

El siguiente paso es [definiendo un manejador de evento](#) para el componente TButton. Usted puede definir manejadores de evento para su aplicación FireMonkey iOS en la misma manera que define manejadores para aplicaciones de escritorio. Para el componente TButton, el evento más típico es un click de botón.

Doble-click al botón sobre el Form Designer, y RAD Studio creará el esqueleto de código que puede usar para implementar un manejador de evento para el evento click del botón:



```
procedure TForm1.Button1Click(Sender: TObject);
30 begin
31 |
- end;
- end.
```

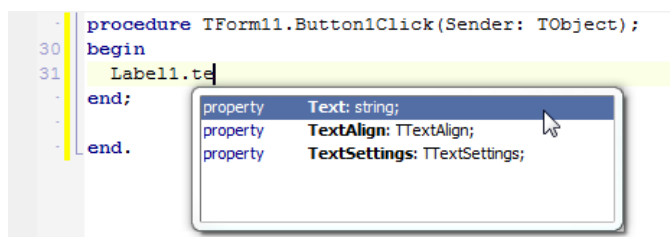
Ahora puede implementar respuestas dentro de las sentencias `begin` y `end` del método **Button1Click**.

El siguiente fragmento de código implementa una respuesta que muestra una pequeña caja de diálogo, que dice "Hola + <nombre ingresado en el edit box>":

```
Label1.Text := 'Hola ' + Edit1.Text + ' !';
```

En Delphi, las comillas que rodean cadenas literales deben ser comillas simples rectas (de ésta manera, 'cadena'). Puede usar el signo más (+) para concatenar cadenas. Si necesita una comilla dentro de una cadena, puede usar dos simples comillas consecutivas dentro de una cadena, que produce una simple comilla.

Mientras escribe código, algunas **tooltip hints (sugerencias)** aparecen, indicando el tipo de parámetro que usted necesita especificar. El tooltip hints también indica el tipo de parámetros que son soportados por una clase determinada:




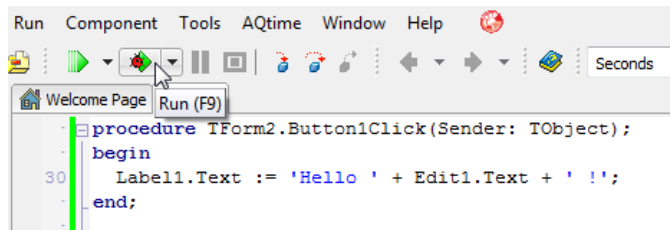
```
procedure TForm1.Button1Click(Sender: TObject);
30 begin
31 Label1.Te
- end;
- end.
```

property Text: string;
property TextAlign: TTextAlign;
property TextSettings: TTextSettings;

Paso 4: Pruebe Su Aplicación Móvil

La implementación de ésta aplicación a finalizado, ahora usted puede ejecutar la aplicación.

Puede hacer click en el botón **Run** () en el IDE, presionar **F9**, o seleccionar **Run** > **Run** desde el menú principal de RAD Studio:

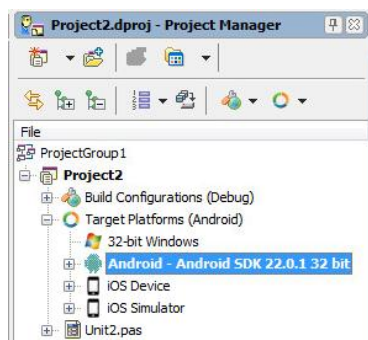


Testeando Su Aplicación Android

Testeando sobre el Android Emulator

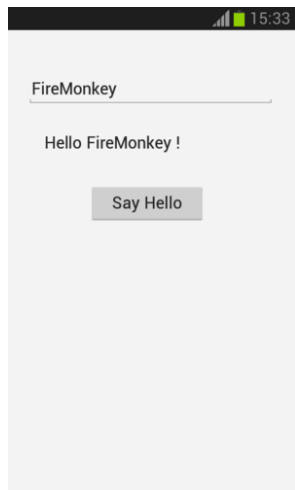
Por defecto, las apps Android corren sobre el **Android Emulator** el mismo que es proveído en RAD Studio.

Antes de ejecutar su app Android, confirme la plataforma de destino en el **Project Manager**:



Testeando sobre el Dispositivo Android

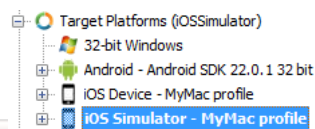
Si completa los pasos descritos en **Tutorial Móvil: Configurando Su Entorno de Desarrollo sobre Windows PC (Android)** antes de crear su nuevo proyecto, puede ejecutar su app Android sobre un dispositivo Android conectado a su PC mediante un cable USB.



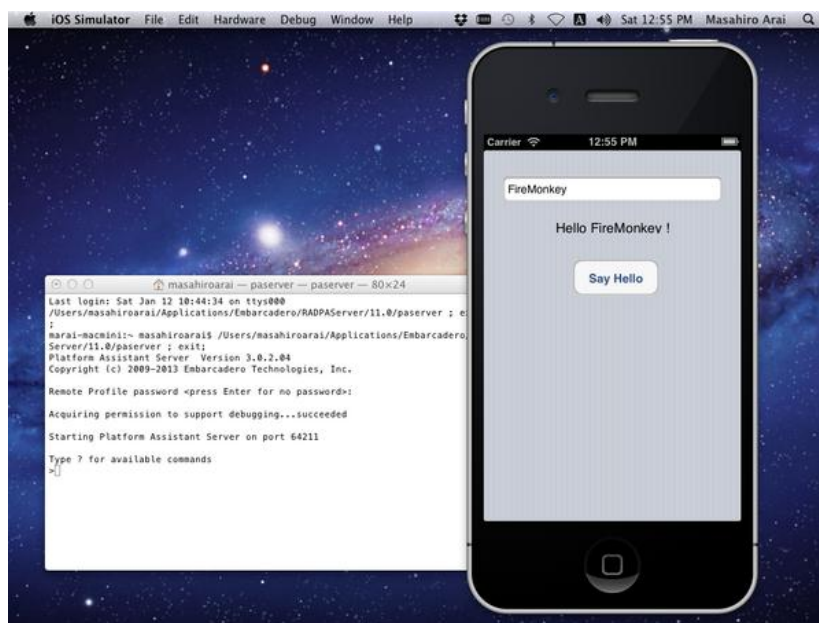
Testeando Su Aplicación iOS

Testeando sobre la Mac (iOS Simulator)

Por defecto, las aplicaciones FireMonkey iOS se ejecutan sobre la plataforma destino **iOS Simulator**. Puede confirmar la plataforma destino en el [Project Manager](#):



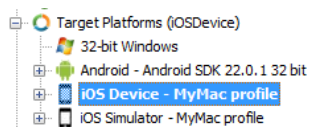
Cuando ejecute su aplicación, ésta es desplegada en la Mac y luego al iOS Simulator sobre la Mac. Para nuestra app, un formulario con una caja de edición y un botón es mostrada. Ingrese un texto en la caja de edición, y click en el botón **Say Hello**:



Testeando sobre un Dispositivo iOS conectado

Si ha completado los pasos descritos en [Tutorial Móvil: Configurando Su Entorno de Desarrollo sobre la Mac \(iOS\)](#) and [Tutorial Móvil: Configurando Su Entorno de Desarrollo sobre Windows PC \(iOS\)](#) antes de crear su nuevo proyecto, ahora puede ejecutar su app iOS sobre un dispositivo iOS conectado a su Mac mediante un cable USB.

Para ejecutar su app iOS sobre un dispositivo iOS conectado, primero seleccione la plataforma destino **iOS Device** entonces el Platform Assistant despliega la aplicación al Dispositivo iOS conectado:



Después que ha seleccionado la plataforma destino **iOS Device**, ejecute su app iOS haciendo click en el botón **Run** en el IDE, presionando **F9** o seleccionando **Run > Run**.

Sobre su Mac, puede ver una caja de diálogo solicitándole permiso para firmar su app iOS. Seleccione "Always Allow" o "Allow" para firmar su app.



Luego vaya a su dispositivo iOS y espere a que su app FireMonkey iOS aparezca. Esté atento a la imagen de lanzamiento de FireMonkey (el ícono está disponible en $\$(BDS)\bin\Artwork\iOS$, puede cambiar la imagen de lanzamiento en [Application Options](#)):

