



Desarrollo de aplicaciones móviles

Ejercicios prácticos

GeneXus 15

Actualizado según upgrade 9

Copyright © GeneXus S.A. 1988-2018.

Todos los derechos reservados. Este documento no puede ser reproducido en cualquier medio sin el consentimiento explícito de GeneXus S.A. La información contenida en este documento es para uso personal únicamente.

Marcas Registradas

GeneXus es marca registrada de GeneXus S.A. Todas las demás marcas mencionadas en este documento son propiedad de sus respectivos dueños.

DESARROLLO DE APLICACIONES MÓVILES	1
EJERCICIOS PRÁCTICOS	1
1. PRIMEROS PASOS	4
2. APLICACIÓN WEB Y DOMINIOS SEMÁNTICOS	4
3. APLICANDO EL PATTERN WORK WITH FOR SMART DEVICES.....	6
Ejecutando la aplicación en forma Compilada	13
4. DESARROLLANDO LA APLICACIÓN	14
Diseño – Themes e Imágenes	14
Personalizando los Objetos	17
Multiples Layouts per Row	21
Agregando nuevas Transacciones (Sessions, Tracks, Rooms, Restaurants)	22
Transacción Restaurant	26
Personalizando el Detail	28
Personalizando las Pantallas de View y Edit	29
Modificando el Estilo de Navegación (Navigation Style)	32
Transitions	33
Órdenes y filtros	34
Comportamiento	35
Integración con otras apps del dispositivo	36
Opciones de invocación: Popup	37
Paneles para ver los Tweets del hashtag del Evento	37
6. ACTIVACIÓN DEL GAM	39
Configuración de la Seguridad a nivel de Objeto	43
7. CONVIRTIENDO LA APLICACIÓN A OFFLINE.....	45

1. Primeros Pasos

A lo largo de este práctico se desarrollará una aplicación para un Evento (en inglés: EventDay).

Se prototipará en Android, dado que se cuenta con un emulador. Si se tiene algún dispositivo (con Android o iOS) podrá probar la aplicación en él.

- Crear una nueva **Knowledge Base** (si usa la versión Trial se creará con Environment .NET con generador Default C# Web)
- Observar las propiedades del generador Default (C# Web —usando la versión Trial encontrará el nodo Generators en el propio KB Explorer—), en particular las del grupo **Execution**.
- Utilizar la opción **Deploy to Cloud** (propiedad del Generador: **Deploy to Cloud = Yes**) —usando la Trial es la opción default y la única posible—.
- Observar las propiedades del **DataStore Default** que indica dónde se va a crear la aplicación —en la versión Trial esta información no está visible—.
- Observar el nodo References y lo que contiene. También los dominios.

El archivo EventDay_BeforeGAM.xpz contiene todos los objetos de la KB solución de este práctico hasta el fin del punto 5 (es decir, antes de la aplicación de GAM).

2. Aplicación Web y Dominios Semánticos

- Crear la Transacción Country de acuerdo a la siguiente figura:

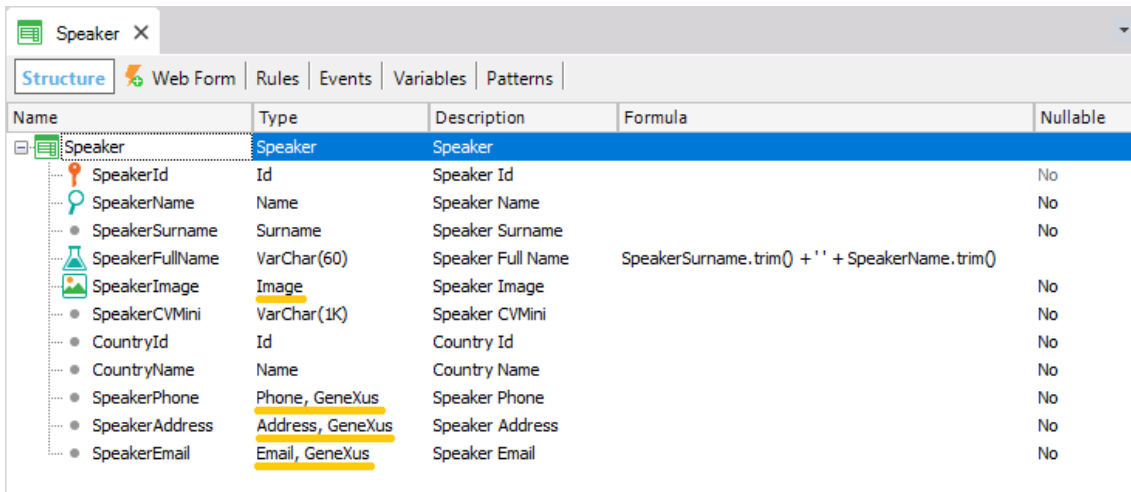
Name	Type	Description	Formula	Nullable
Country	Country	Country		
CountryId	Id	Country Id		No
CountryName	Name	Country Name		No
CountryFlag	Image	Country Flag		No

Crear los siguientes Dominios:

- Id - Numeric(8.0)
- Name - Varchar(20)

Configurar en el atributo CountryId la propiedad Autonumber (en el atributo, no en el dominio).

- Crear la Transacción **Speaker** de acuerdo a la siguiente figura:



Name	Type	Description	Formula	Nullable
Speaker	Speaker	Speaker		
SpeakerId	Id	Speaker Id		No
SpeakerName	Name	Speaker Name		No
SpeakerSurname	Surname	Speaker Surname		No
SpeakerFullName	VarChar(60)	Speaker Full Name	SpeakerSurname.trim() + ' ' + SpeakerName.trim()	
SpeakerImage	Image	Speaker Image		No
SpeakerCVMini	VarChar(1K)	Speaker CVMini		No
CountryId	Id	Country Id		No
CountryName	Name	Country Name		No
SpeakerPhone	Phone, GeneXus	Speaker Phone		No
SpeakerAddress	Address, GeneXus	Speaker Address		No
SpeakerEmail	Email, GeneXus	Speaker Email		No

- Configurar la propiedad Autonumber en el atributo SpeakerId
- Definir el dominio Surname = VarChar(40)
- SpeakerFullName es un atributo fórmula = SpeakerSurname.trim()+ ' ' + SpeakerName.Trim()



Observar los dominios predefinidos en GeneXus. Varios corresponden a dominios semánticos. ¿Qué significa esto?

- Agregar reglas:

```
Error( 'The Speaker Name must not be empty')
    if SpeakerName.IsEmpty();

Error( 'The Speaker Surname must not be empty' )
    if SpeakerSurname.IsEmpty();

Msg( 'Curriculum Vitae should not be empty')
    if SpeakerCVMini.IsEmpty();
```

- Aplicar a ambas Transacciones el Pattern **Work With for Web**
- Importar en la KB el archivo: **1-InitializeCountries_and_Speakers.xpz**. Aquí básicamente importará dos procedimientos para inicializar las tablas con datos (se incluyen las imágenes de banderas de países y fotos de speakers). Para ello:

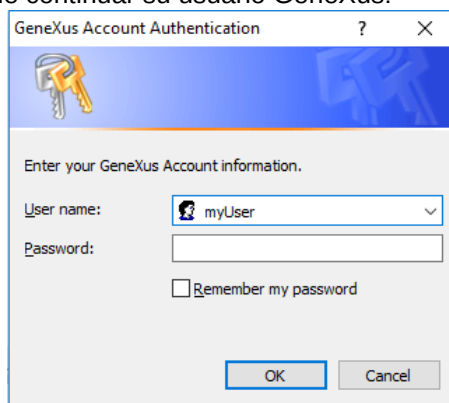
Agregar en el Evento Start del Web Panel **Home** (se encuentra en el folder GeneXus/Web) la invocación a los siguientes procedimientos que se importaron con el xpz anterior:

```
Event Start
    InitializeCountries()
    InitializeSpeakers()
Endevent
```

Nota importante: se desarrollaron dos procedimientos para crear los datos para no tocar las transacciones que usted creó. En GeneXus 15 la forma natural de inicializar con datos

las tablas de ambas transacciones es utilizar la propiedad “Data Provider” de la transacción, junto con Used to “Populate Data”. (Para interiorizarse en este tema, puede ver el video: <http://training.genexus.com/genexus/curso-genexus-15?es#poblar-con-datos-desde-la-propia-transaccion>)

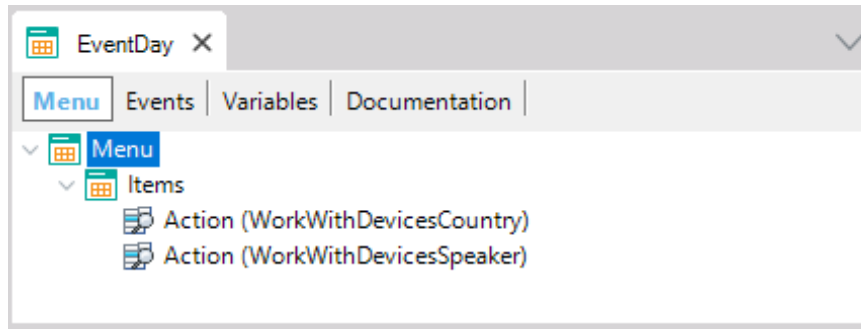
- Ejecutar la aplicación presionando F5 (tendrá que aceptar la creación de las tablas de la base de datos). Por estar prototipando en la nube, si en la sesión actual no lo había hecho antes, se le pedirá antes de continuar su usuario GeneXus:



- Ejecutar el Web Panel **Home** para la inicialización y ver los datos cargados. Acceda al **Work With Countries** y al **Work With Speakers** para comprobar que efectivamente se hayan cargado.

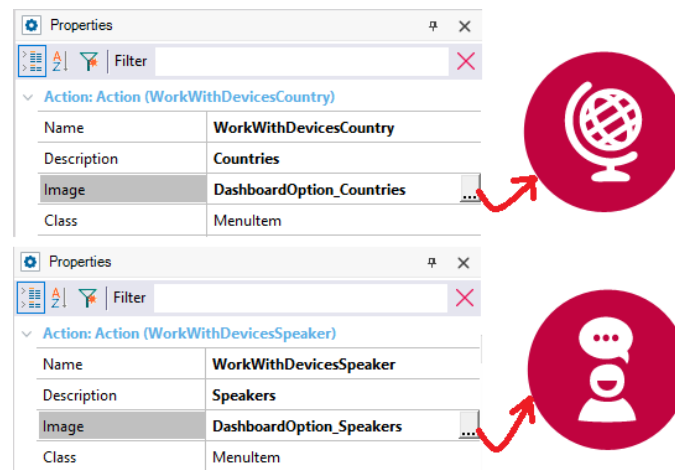
3. Aplicando el Pattern Work With for Smart Devices

- Aplicar el Pattern **Work With for Smart Devices** a las dos transacciones.
- Observar los nodos **List**, **Detail** y **Section(General)** del objeto Work With for Smart Devices, viendo en cada caso la pantalla que aparece.
- Grabar y observar qué sucede con los siguientes elementos:
 - Propiedades de la transacción:
 - **Business Component** = True
 - **Expose as Web Service** = True
 - **Web Services Protocol** = ReST Protocol
 - Observar cómo aparece el objeto Work With for Smart Devices en el KB Explorer, bajo la transacción.
 - Se agregó el generador para **Smart Devices** como generador secundario.
- Crear el Menu for Smart Devices **EventDay** (respetar el nombre para las importaciones de objetos posteriores), de acuerdo a la siguiente figura:



Observar el código del evento asociado a cada opción. En el primer ítem se está invocando al Work With de países. ¿A qué parte del mismo? Aprecie la sintaxis de la invocación.

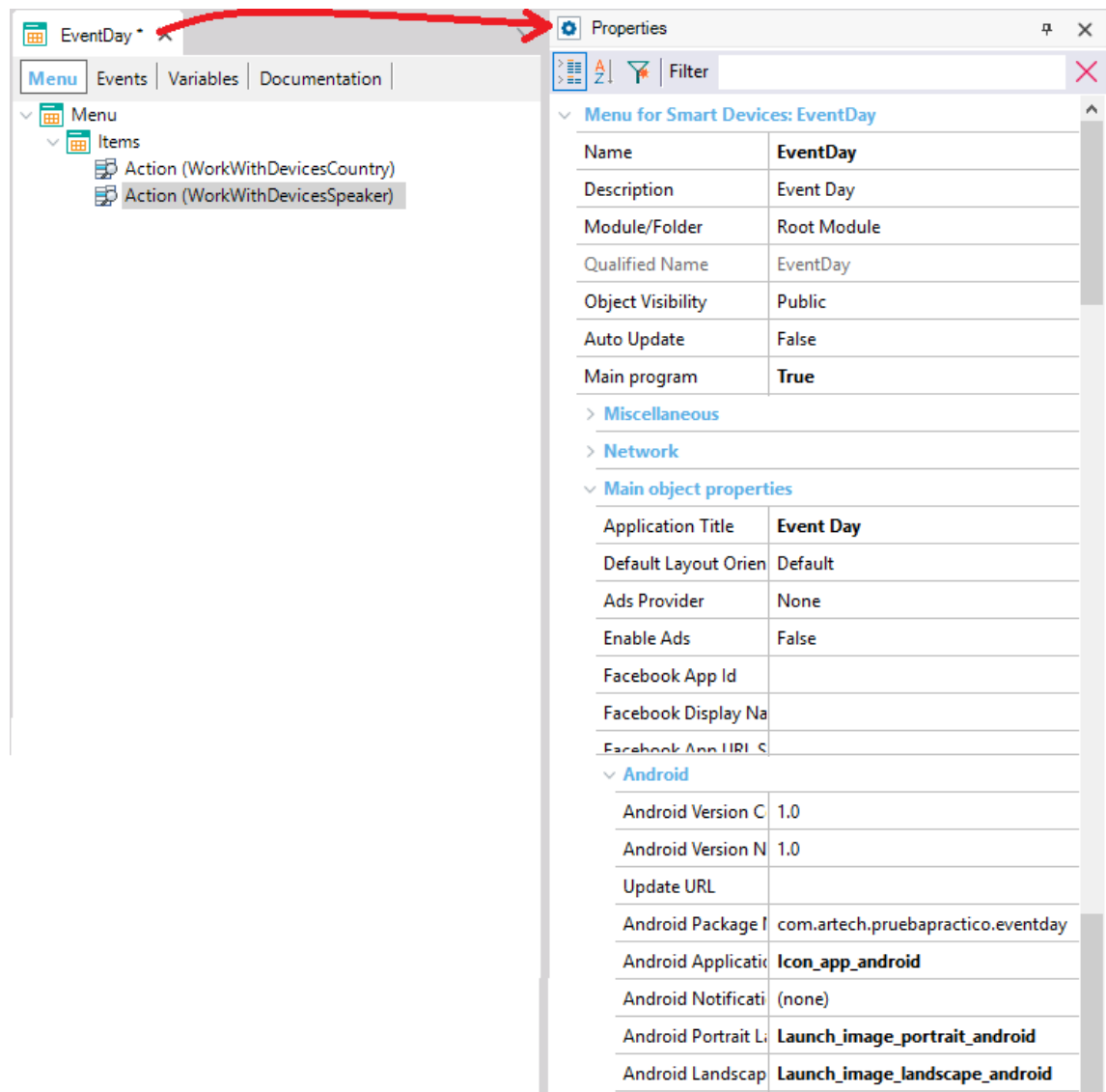
- Asociar una imagen a cada opción (provistas junto con este material):



- Aproveche, también, para editar las propiedades del menú EventDay, ponerle el título a la aplicación y defina la imagen de la app y las de launch.



¿Por qué las define en este objeto?



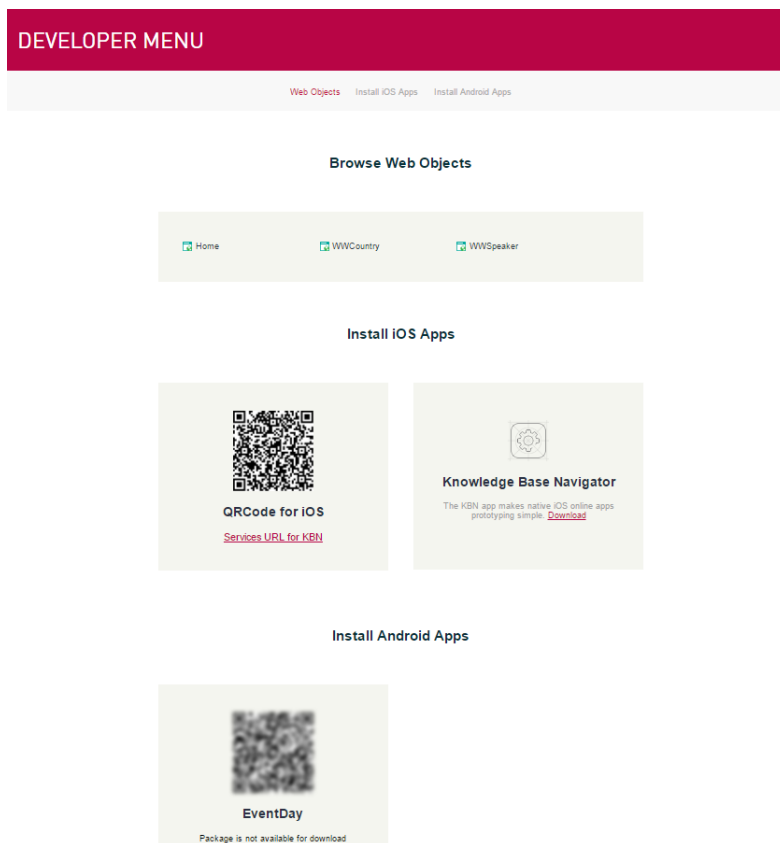
- Grabar y observar que la propiedad **Main Program** está en “True”.
- Presionar **F5** para generar y ejecutar la aplicación tanto web como para Android.

Consecuencias:

- Dado que no se ha especificado ningún “Startup Object”, GeneXus abrirá el Developer Menu web en el navegador predeterminado de su computadora.

Nota: si tiene un dispositivo iOS y quiere probar allí, instálele el KBN (observe entre los QR codes que aparecen en el Developer Menu, el que es para descargar el KBN), ejecútelo, y mediante el menú, elija la opción “Add Knowledge Base” y luego “+”. Escanee el QR code que aparece en el Developer Menu para el KBN (el único que aparece habilitado). Grabe y pruebe.

- En el **Developer Menu**, además de los links para ejecutar los objetos de la aplicación Web, aparecerán QR codes. Recuerde que en el teórico vio para qué se utiliza cada uno.
- No siempre se creará el archivo compilado. Dependerá de si se tiene “Startup object” configurado y de la modalidad de ejecución que se empleó (si F5, Run o “Run with this only” sobre un objeto main, etc.) En nuestro caso aún **no tenemos el archivo compilado**, por lo que el QRcode para Android aparece deshabilitado:



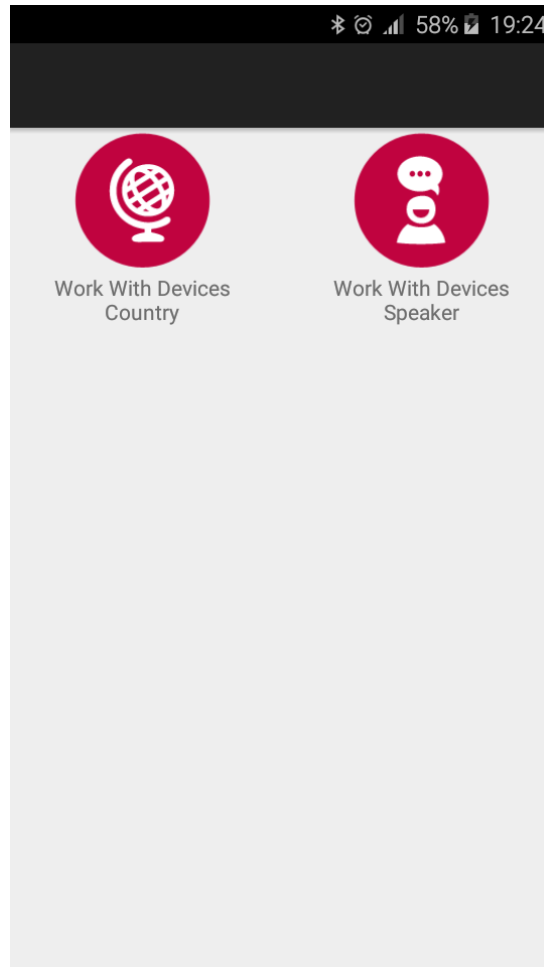
- Pero como está ejecutando para Android, que **no admite** ejecutar con el KBN, se le mostrará un error:

```
===== Android Execution started =====
error: A startup object must be selected
Android Execution Failed
Run Developer Menu Failed
```

- Sobre el objeto menú **EventDay** haga botón derecho y “Set as Startup object”.

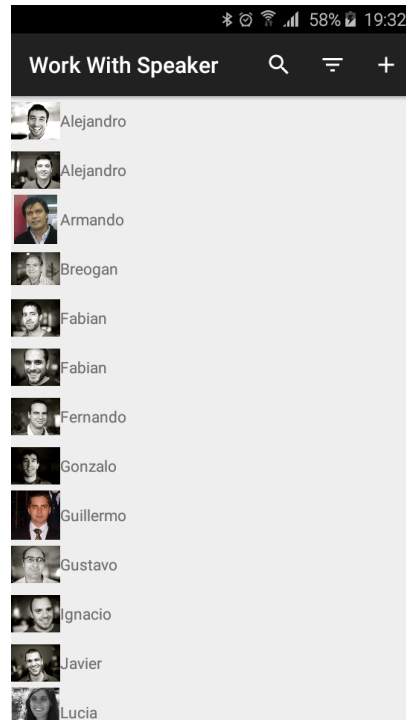
 Verifique que el checkbox del Theme CarmineSD esté marcado y si no, hágalo antes de ejecutar.

- Ahora sí, F5, para ejecutar ese objeto y todas sus dependencias. Aquí sí se compilará y si refresca la página del Developer Menu web, verá que ahora sí aparece el QR code habilitado. Verá que GeneXus abre automáticamente el emulador de Android y en el mismo vemos que se ejecuta el **menú** de la aplicación.

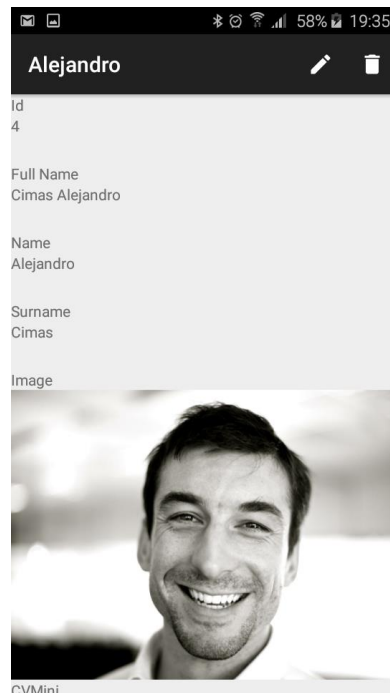


En el emulador:

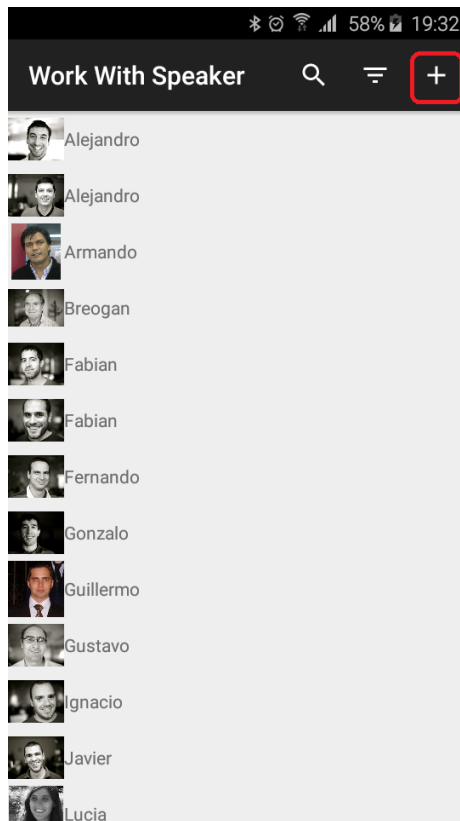
- Hacer *tap* sobre “**Speakers**”
 - Observar que aparecen los mismos oradores que se ingresaron desde la aplicación Web
 - Observar los atributos que se están mostrando de cada orador. Ir a GeneXus a buscar en el Pattern el Layout de este **List**, para ver la correspondencia.



- Haga *tap* sobre un orador cualquiera, y vea cómo aparece el detalle del mismo (¿este es el **Layout** correspondiente al modo **View** o al **Edit**?):



- **Editar** el orador que está visualizando. Al hacerlo, ¿qué **Layout** se está desplegando para esta **Sección General** del detalle? ¿El del modo **View** o **Edit**?
- **Insertar** un nuevo Speaker. Para ello:

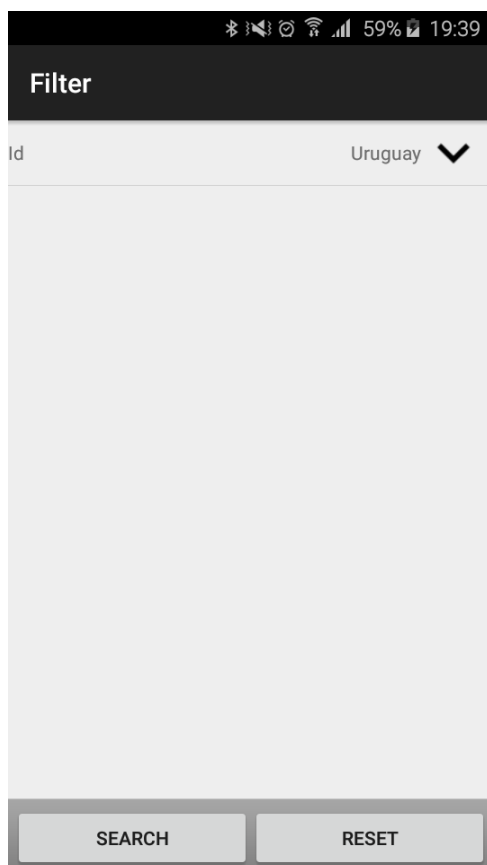


- Intentar dejar vacío el nombre o el apellido. ¿A quién se invoca cuando se presiona la imagen a la derecha de CountryId?



Grabar. ¿Por qué no le permite realizar la operación? Ingresar el nombre o apellido y volver a probar. Observar cómo aparece el nuevo orador en la lista.

- Pruebe hacer búsquedas de oradores por nombre. Luego pruebe por apellido. ¿Por qué otros datos puede hacer searches? Vaya al Work With a observar dónde están configuradas esas búsquedas. Elimine alguna si no le parece conveniente y vuelva a probar.
- Pruebe filtrar por el país:

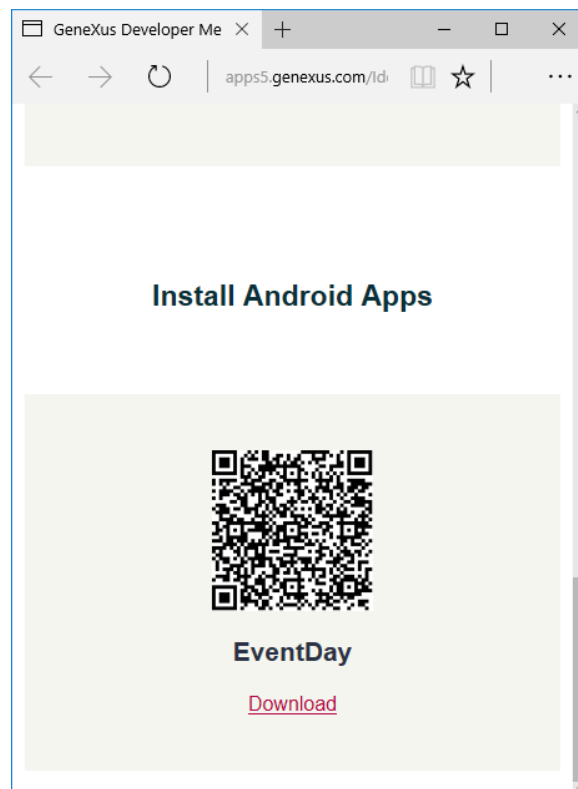


Ejecutando la aplicación en forma Compilada

Como hemos dicho, en Android ya no es posible interpretar la aplicación a través del KBN. Para obtener y ejecutar el compilado existen varias alternativas:

- Hacer botón derecho sobre la pestaña del **Menu for Smart Devices** y elegir la opción “**Run With this Only**” o “**Run**”.
- Especificar el **Menu for Smart Devices** como **Startup Object** (botón derecho sobre la pestaña del objeto/Set As Startup Object, o Preferences/.Net Environment, propiedad “Startup Object”, si no está utilizando la versión Trial). En el próximo F5 se compilará sólo la aplicación que depende de este objeto, es decir, en este caso, sólo la aplicación para Smart Devices cuyo punto de entrada es el Menu for Smart Devices. No abrirá el Developer Menu Web, sino sólo el emulador con el Menu for Smart Devices compilado ejecutándose (si tiene el dispositivo conectado a la computadora, se abrirá directamente en el dispositivo).
- Existen más alternativas. Sugerimos buscarlas en wiki.genexus.com o aquí, donde están resumidas y explicitadas: <http://training.genexus.com/genexus/curso-genexus-15-analista?es#proceso-de-build>.

Recordar que al compilar aparecerá en el Developer Menu web el **QR Code** correspondiente. Puede obtenerlo mediante **View / Show QR Codes**:

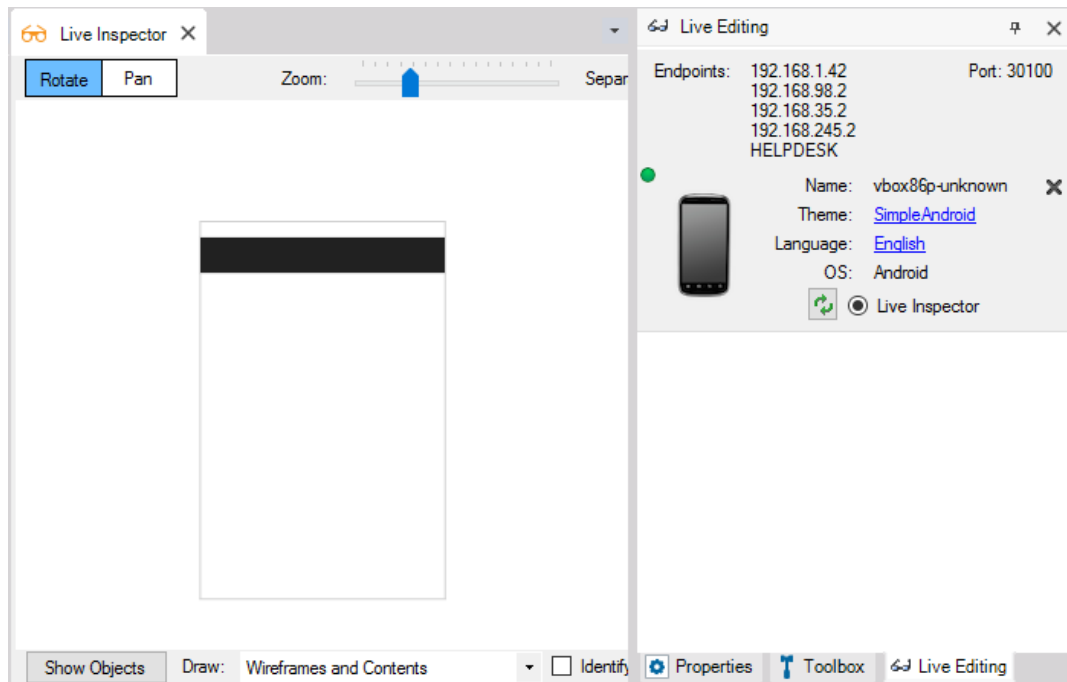


Si cuenta con dispositivo Android pruebe a través de su programa de lectura de QR Codes descargarse e instalar la aplicación compilada. Aunque la mejor opción será sin duda conectar con cable el dispositivo a la computadora que está ejecutando GeneXus.

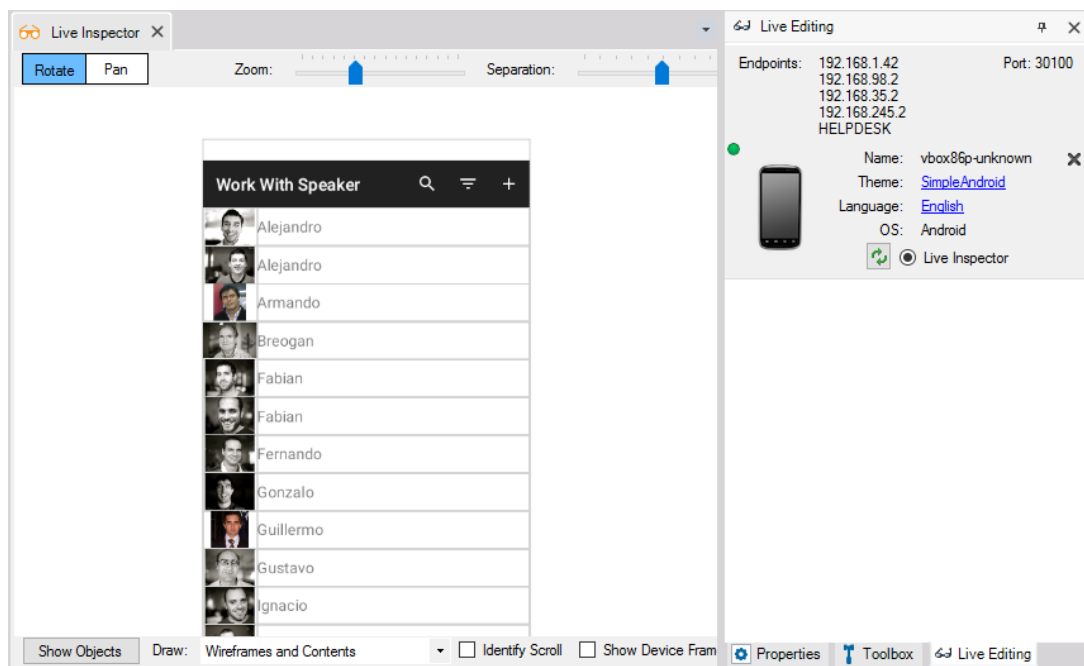
4. Desarrollando la Aplicación

Diseño – Themes e Imágenes

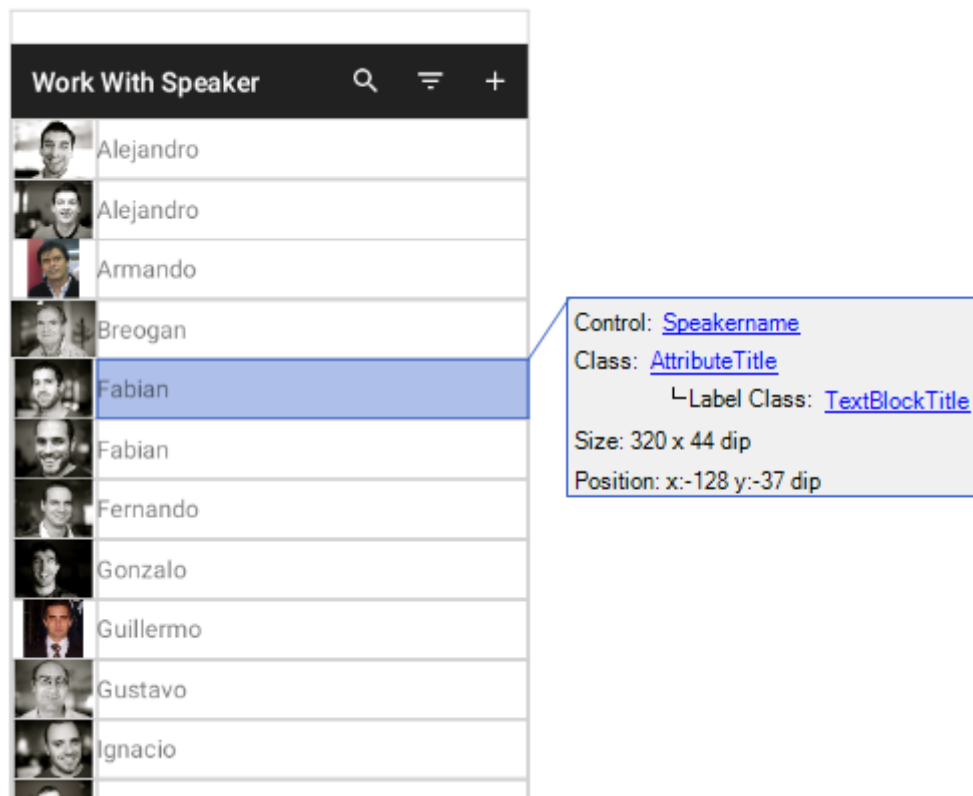
- Observe los themes que tiene en su KB, y qué theme se utiliza para cada plataforma (para ello, acceda al folder Customization, nodo Themes y SD Platforms → Any Android, Any iOS, etc). Vea que por defecto tienen configurada la propiedad Theme con el valor CarmineSD, CarmineAndroid o CarmineIOS. Para la plataforma AnyAndroid, cámbielo por SimpleAndroid. Si no está marcado el check box de SimpleAndroid (en Themes/SimpleAndroid). márquelo para que se importe en la KB.
- Habilite el Live Editing (a través del combo en la barra superior en GeneXus). Vea la ventana de ese nombre y haga un nuevo Run sobre el objeto Menu for Smart Devices, para que Live Editing empiece a “escuchar”.
- Observe cómo aparece la entrada en la ventana de Live Editing, y se abre la ventanade Live Inspector, que le muestra la app que está corriendo en su emulador o dispositivo:



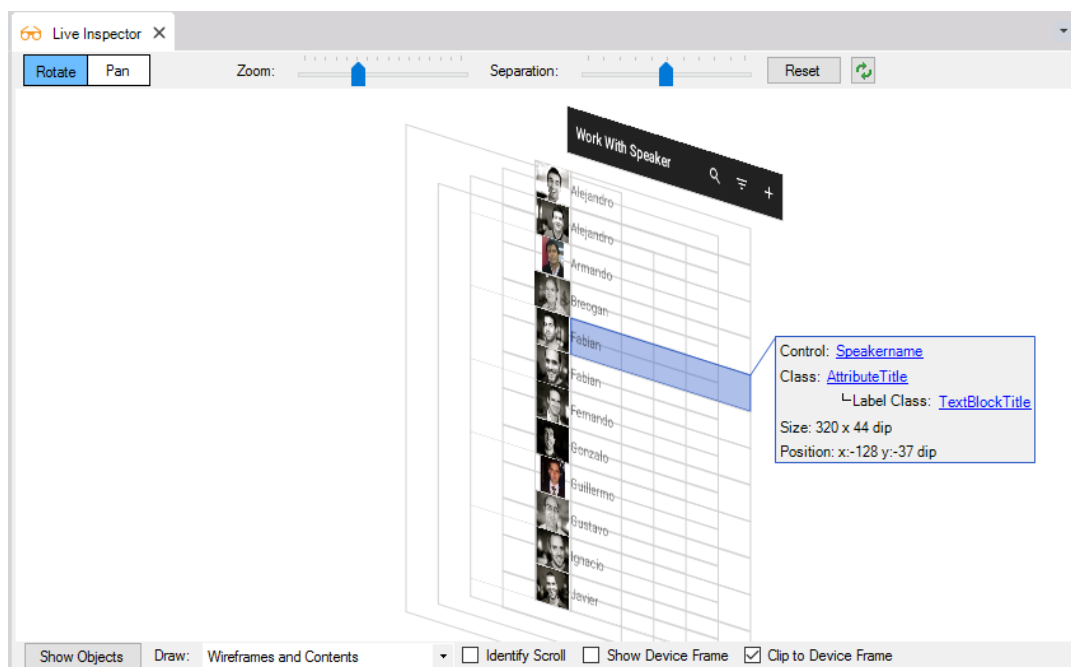
Nota: el Menu for Smart Devices no es inspeccionable. Sólo lo son los Work Withs y los Panels. Pruebe entrar al List de Speakers, y ahora sí vea la pantalla de Live Inspector:



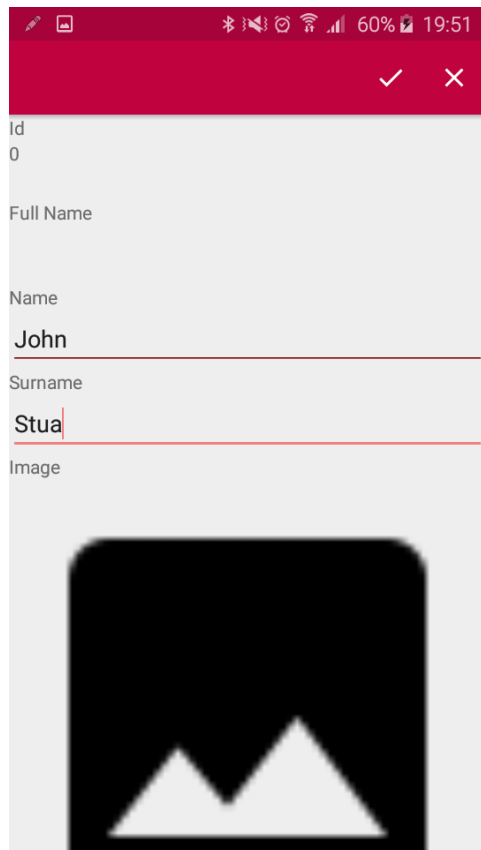
Observe que si se posiciona sobre el nombre de un orador del grid podrá ver el control del que se trata (el atributo SpeakerName), así como su clase, tamaño y posición:



- Rote con el mouse la pantalla para ver las capas:



- Se desea que la Application Bar asuma el color #C0033F y la Status Bar el color #A50339. Además, que el color cuando los campos en edición se muestran en la pantalla pero sin ser el campo activo, aparezcan subrayados en color Maroon, y cuando se trata de un campo activo, en LightCoral:



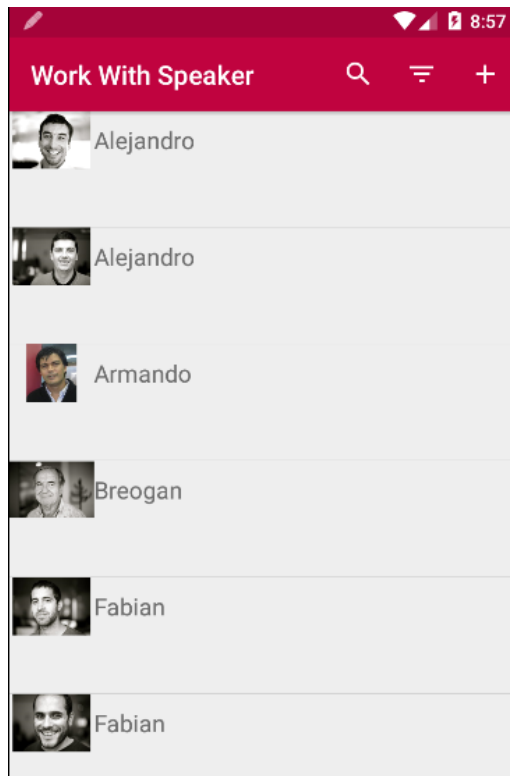
Para ello: edite las propiedades de la clase Application del theme. Es posible que Live Editing no reconozca estos cambios. En ese caso haga Run sobre el objeto main (o F5 si lo tiene marcado como Startup object) para verlos aplicados.

- Observe en el Theme SimpleAndroid las subclases de la clase Attribute y también observe las subclases de la clase Table. Luego importe el archivo **2-ThemeSimpleAndroid.xpz**

Este xpz contiene únicamente el theme SimpleAndroid con nuevas clases que se han creado para que usted no tenga que hacerlo. Por ejemplo, observe las clases AttributeBrandColored, AttributeTitle_BrandColor, y AttributeTracks, junto con sus subclases. Observe la TableTrack, subclase de Table, y todas sus subclases, que son las que permitirán dar color a las filas del grid de conferencias, luego. También se han definido las subclases de Image: ImageFill, ImageFillKeepingAspectRatio, ImageFit.

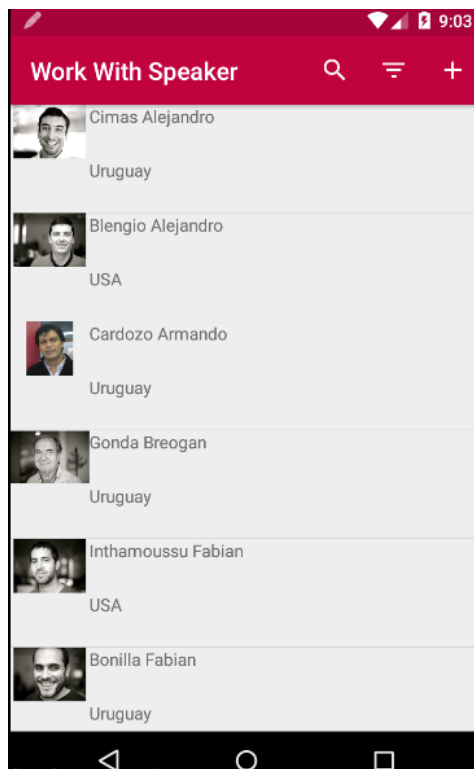
Personalizando los Objetos

- Asegurarse de estar inspeccionando con Live Editing su app. Agregar al **Grid** que muestra la lista de Speakers el país del orador (sin etiqueta). Observe que en su dispositivo o emulador aparecerá el espacio en blanco, ¿Por qué?

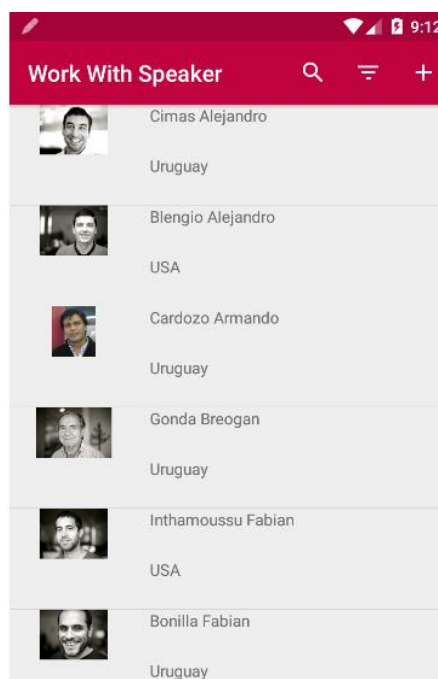
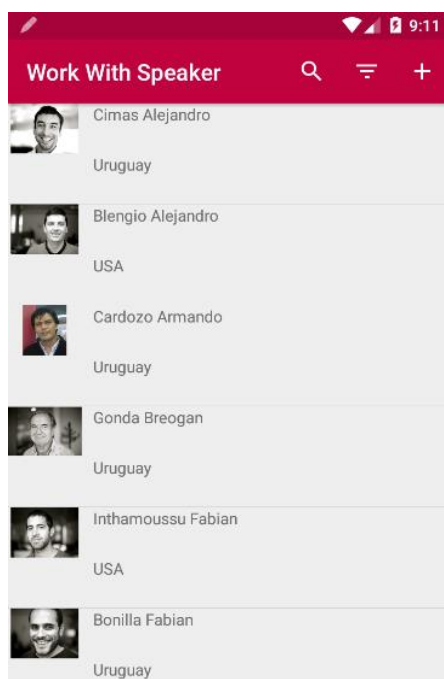


Respuesta: Live Editing utiliza la metadata para mostrar los resultados en el dispositivo y en la ventana Live Inspector sin compilar, en forma instantánea. Al agregar un control atributo, su contenido debe buscarse en la base de datos, a través del servicio rest (data provider) correspondiente y por lo tanto el Live Editing no podrá actualizarse y se mostrará un mensaje de que la conexión fue interrumpida. Para ver los cambios y que el LiveEditing se resincronice, deberá ejecutarse nuevamente el objeto main

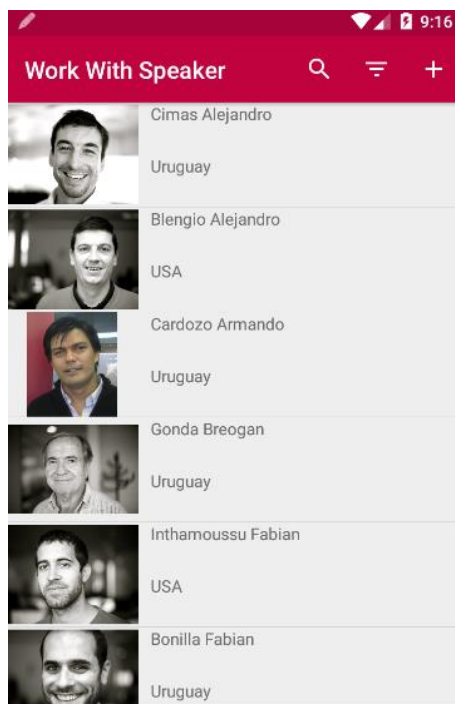
- Observar la clase asociada al control Att/Var SpeakerName. Borre el atributo SpeakerName y ponga en su lugar al atributo SpeakerFullName, también sin etiqueta. Observe dos cosas: 1) que la clase asociada al nuevo atributo será Attribute, y no AttributeTitle, y 2) que ahora en la ventana de Live Inspector, así como en su dispositivo, no se verán los cambios realizados. Para verlos en ejecución, haga Run sobre EventDay o F5 si es Startup Object.



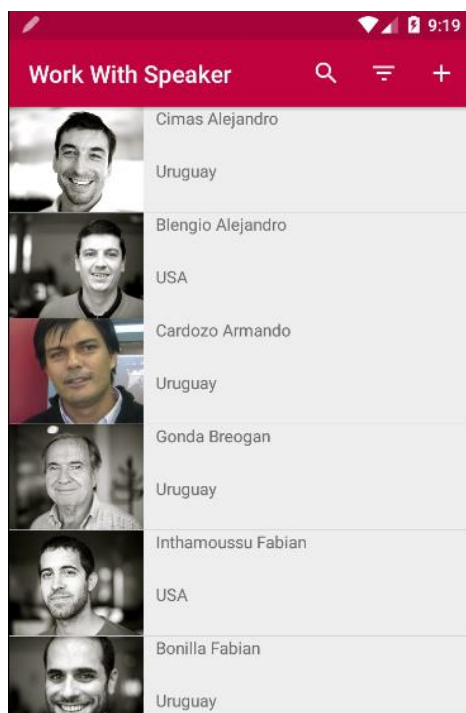
- Hay varias cosas a corregir en esta pantalla: una es la distribución de los controles. No queremos que quede pegada la foto al nombre del orador y su país. Para ello, podemos agregar una columna entre medio, **de 10 dips**, para dar espacio entre los contenidos de la primera columna y la segunda. Live editing permitirá visualizar inmediatamente el cambio. Al hacerlo, notamos que la primera columna está tomando un tamaño fijo de 44 dips (en este ejemplo), cuando en verdad desearíamos que ocupara un 30% del ancho restante (quitando los 10 dips de la columna del medio) –por lo que la tercera tendrá que ocupar el 70% restante:



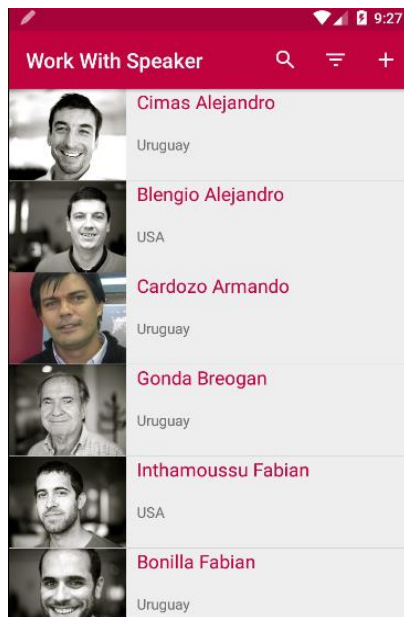
- Ahora queremos que la imagen ocupe dos filas, y no una (propiedad Row span). Observe que al cambiar el valor de esta propiedad, el layout puede desajustarse, haga los arreglos que haga falta.



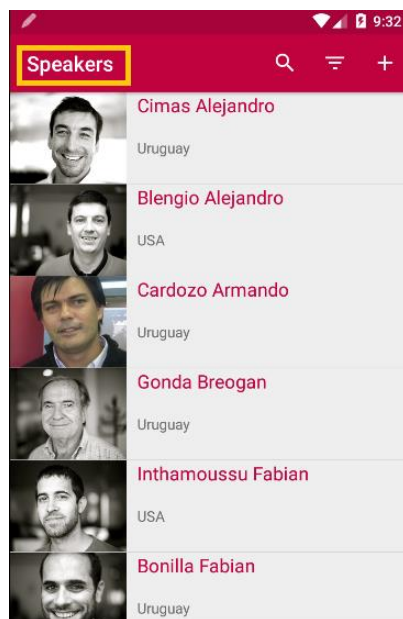
- Vemos que las imágenes están autoajustándose para entrar (fit) completamente en el tamaño de la celda donde se encuentran. Es por esta razón que la foto de Cardozo Armando no está ocupando todo el ancho. No deseamos esto, sino que se escalen en forma proporcional, para ocupar totalmente la celda (recuerde que en la importación del theme se habían agregado las subclases de la clase Image, que configuran la propiedad Scale Type).



- Queremos que el nombre completo del orador aparezca destacado, y en el color principal de la app. Recuerde que en el theme se agregó la clase `AttributeTitle_BrandColor`:



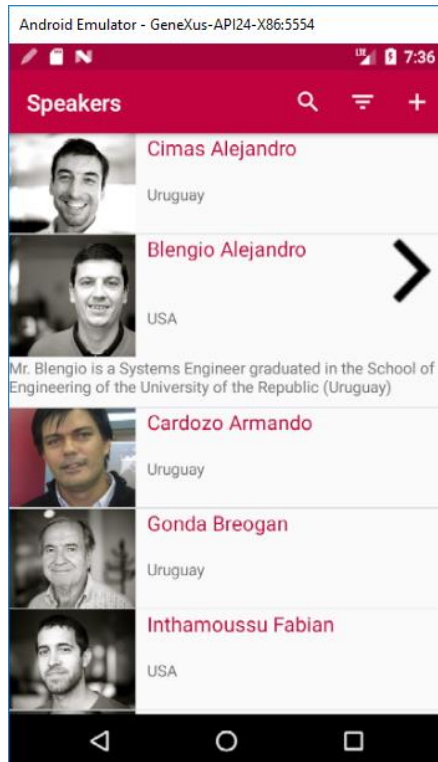
- Observe que en el List aparece el caption “Work With Speaker” arriba. Cámbielo por Speakers (propiedades del nodo List):



Multiple Layouts per Row

- Ahora se quiere que cuando el usuario hace *tap* sobre un Speaker en el **List**, en lugar de ir directamente a mostrar la información detallada del Speaker, se amplíe la propia línea, mostrando allí el **Curriculum Vitae**.

- Para ir al detalle, el usuario deberá hacer *tap* sobre una imagen de **flecha** (arrow_right_black.png, que se encuentra junto con el material que se le entregó)



Habrás que crear otro “ítem layout” para ese grid y especificar cuál debe mostrarse cuando la línea/ítem está seleccionada. Pista: posicóñese sobre el grid, en su extremo superior derecho y haga clic.

Al agregar la imagen con la flecha, en una cuarta columna, deberá redefinir los tamaños de las columnas: 30%;10dip;55%;15%.

Para el CV, le recomendamos que se expanda a las 4 columnas y que pueda crecer (Auto Grow en true ocupando el tamaño que necesite)

Preste atención a las propiedades “Default Action” y “Default Selected Item Layout” del grid. Deberá deshabilitar la invocación al Detail cuando se hace tap sobre la línea, para que en esta acción se cargue el nuevo layout para la fila.

Recuerde que a los controles en el form se les pueden asociar eventos relativos a las acciones que el usuario puede realizar con el dedo sobre ellos (haga botón derecho sobre el control imagen, Go to Event... y vea qué sucede). Recuerde cuál era la sintaxis de las invocaciones al Detail:

(solución: `WorkWithDevicesSpeaker.Speaker.Detail (SpeakerId)`)

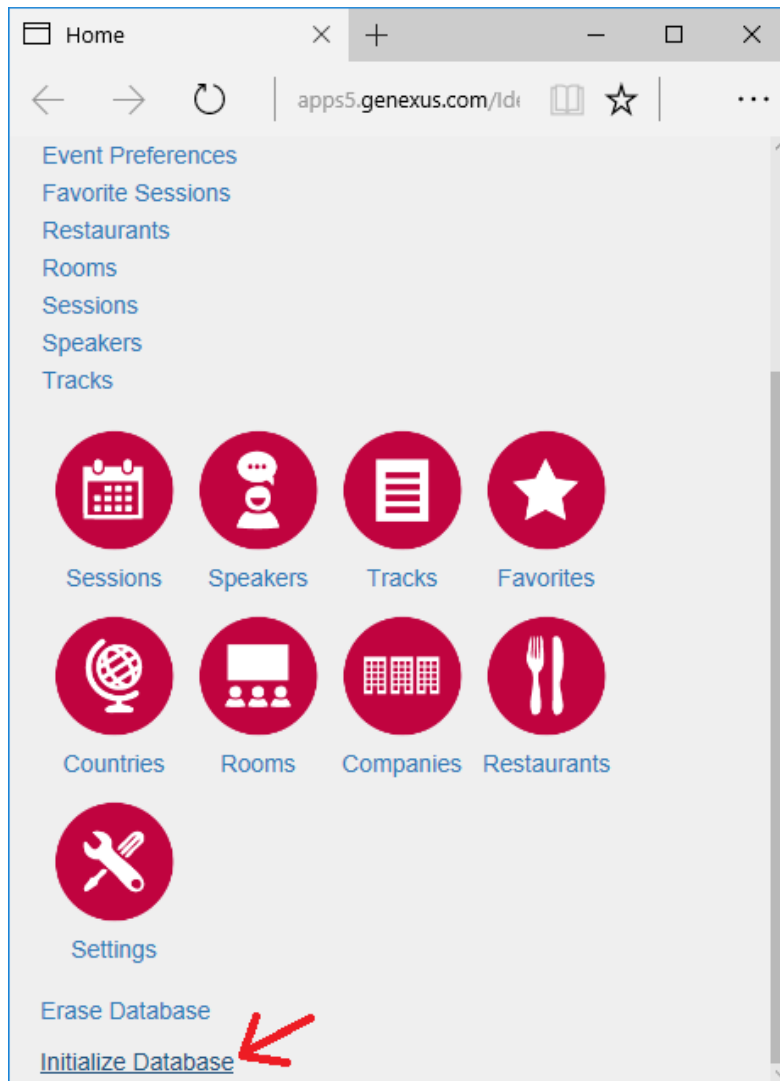
Agregando nuevas Transacciones (Sessions, Tracks, Rooms, Restaurants)

- Importar **3-MoreTransactions.xpz**.

Este xpz contiene, entre otras cosas:

- TRNs: Track, Room, Session, Restaurant, FavoriteSessions, EventPreferences

- Folder SampleData con los data providers:
 - DPREstaurant
 - DPTrack
 - DPRoom
 - DPsession
- Dominios: Description, Discount, Floor, QuickestServiceTime, tec.
- Folder: EventProcsDP con los procedimientos
 - Procedure: GetSessionSpeakers
 - Procedure: GetSessionTracks
 - Procedure: GetEventDates (para obtener fecha inicial y fecha final del evento, aunque no lo utilizaremos en este práctico)
 - Procedure: IsFavoriteSession
 - Procedure: SetFavoriteSession
- Web panel Home
- Imágenes:
 - Sample_restaurant*
 - Sample_room*
 - Clock*
- Menu for Smart Devices: EventDay
- Hacer Run sobre el web panel **Home** (le pedirá reorganizar para que se creen las tablas correspondientes a las nuevas transacciones). En el web panel Home en ejecución, presionar Initialize Database (ese proc inicializará la info de todas las tablas, excepto las de Countries y Speakers, que ya tienen datos).

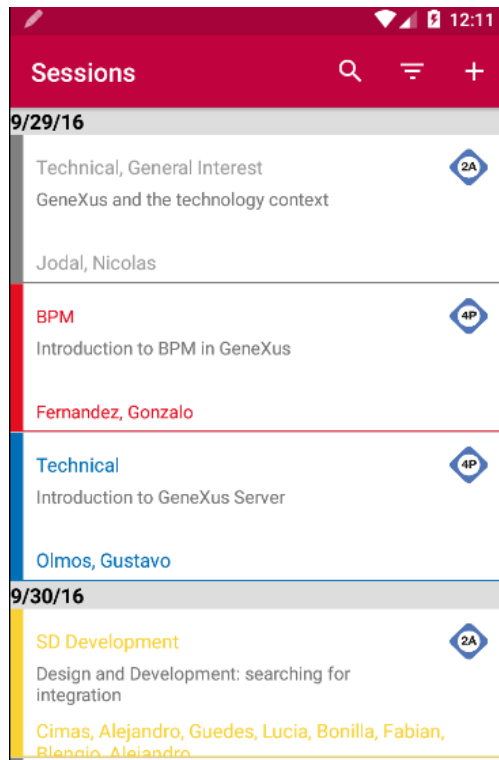


- Tomarse un tiempo para familiarizarse con la estructura de la Transacción **Session** que almacena la información de las conferencias del evento
- Ejecutar el Menu for Smart Devices (F5 si lo tiene como Startup Object), ir a la opción **Sessions** (mostrará el List del Work with)

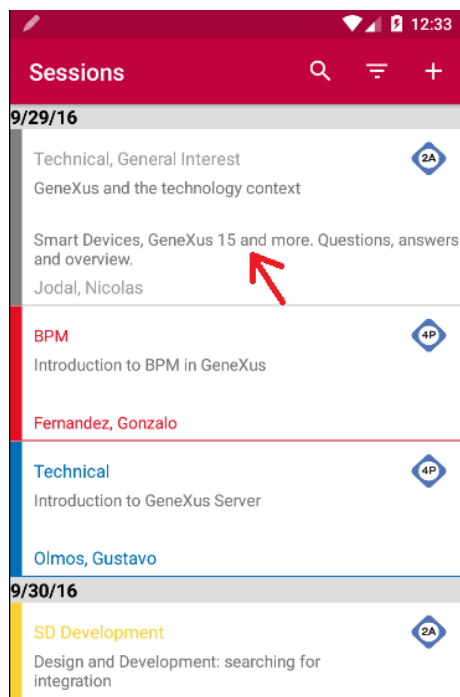


Observar que cada Session se carga con un color determinado. Puede investigar cómo se implementó esto, yendo a la solapa de Events.

- Las Sessions salen **agrupadas por día**. ¿Cómo se consiguió esto?



- Si ahora quiere que cuando la **Session** es una **Keynote** se muestre distinta la información, por ejemplo, **agregando la descripción** (como se ve en la pantalla que sigue, para la primera session “GeneXus and the technology context”), ¿cómo se implementa?

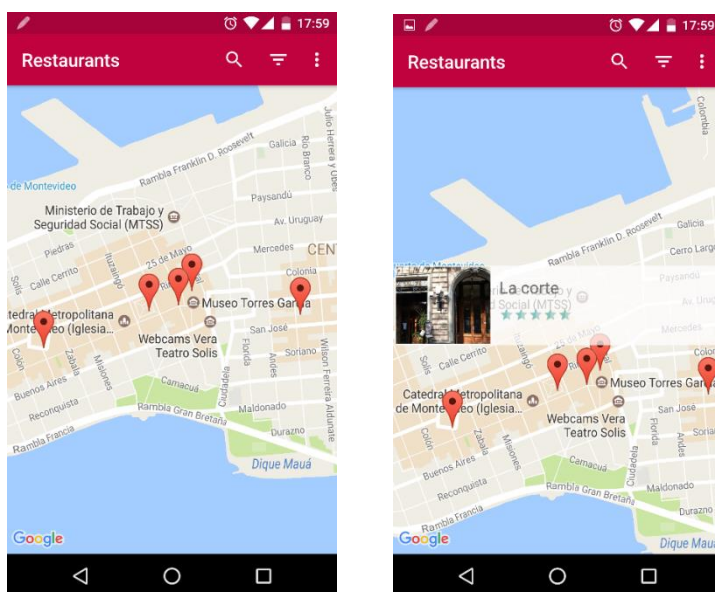


Recuerde la propiedad `ItemLayout` dinámica, del `grid`. Y la `Auto Grow`, estática, para `SessionDescription`. ¿Por que no ve los cambios con `Live Editing`? Observe

que la programación se realiza dentro del evento Load del sistema. Más adelante verá que este evento se ejecuta en el Server, dentro del DP rest correspondiente.

Transacción Restaurant

- Observar la Transacción **Restaurant**. Se quiere que la lista de Restaurants no se muestre de la manera estándar, sino que los restaurantes se muestren **como puntos en un mapa**, ya que contienen un atributo geolocation. Y que al hacer tap sobre un punto, se vea su información, y al hacer tap de vuelta sobre ésta, se muestre su detalle.

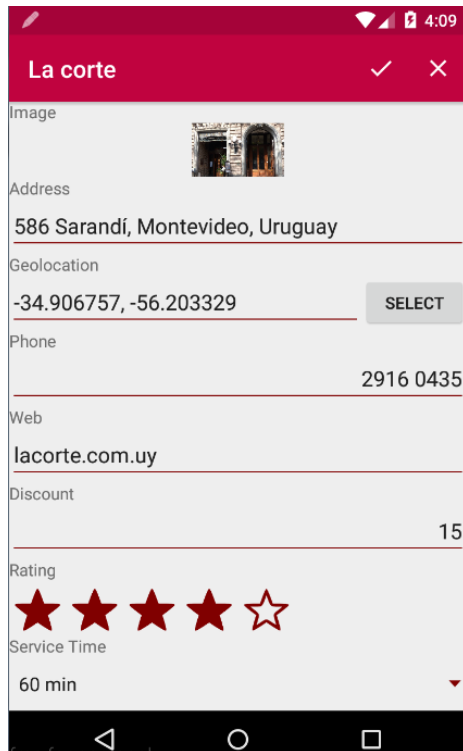


Nota: Para que la app de mapas le funcione, deberá tener instalado en el emulador Google maps o en su dispositivo. Y en cualquier caso, deberá configurar la API Key de Android Maps, a nivel del objeto main (en nuestro caso el menú EventDay). Pídale al instructor esa clave para poder probar:

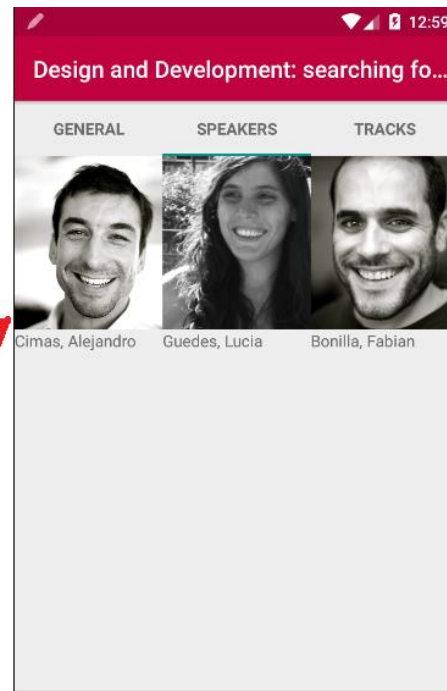
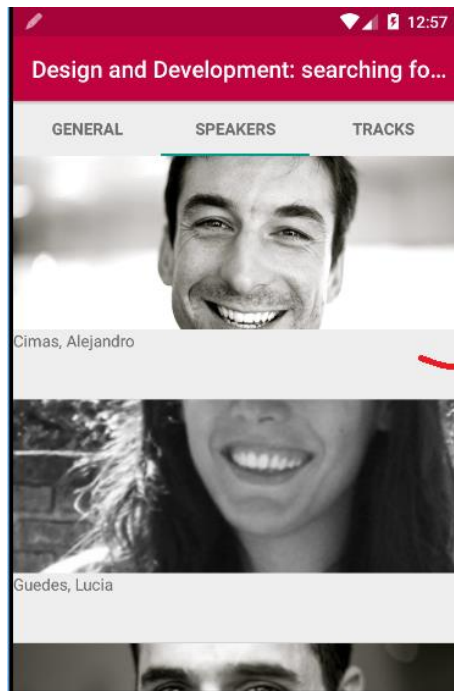
Android Specific	
Android SDK directory	C:\Android-SDK
JDK Directory	C:\Program Files\Java\jdk1.8.0_20
Copy APK to Cloud	True
Multidex Build	False
Gradle Options	--daemon --parallel -Dorg.gradle.jv...
Application Signing	
Key Store File	C:\full15u3\Android\debug.keystore
Key Alias	androiddebugkey
Store Password	*****
Key Password	*****
Android Maps API	Google Maps API v2
Android Maps API Key	
In Application Billing	

- También observe que el rating del restaurant se está mostrando con estrellitas. ¿Cómo consigue esto?

- Si luego va a editar la información de un restaurant, el usuario podrá ingresar también el rating en estrellas. Impleméntelo (con Live Editing).

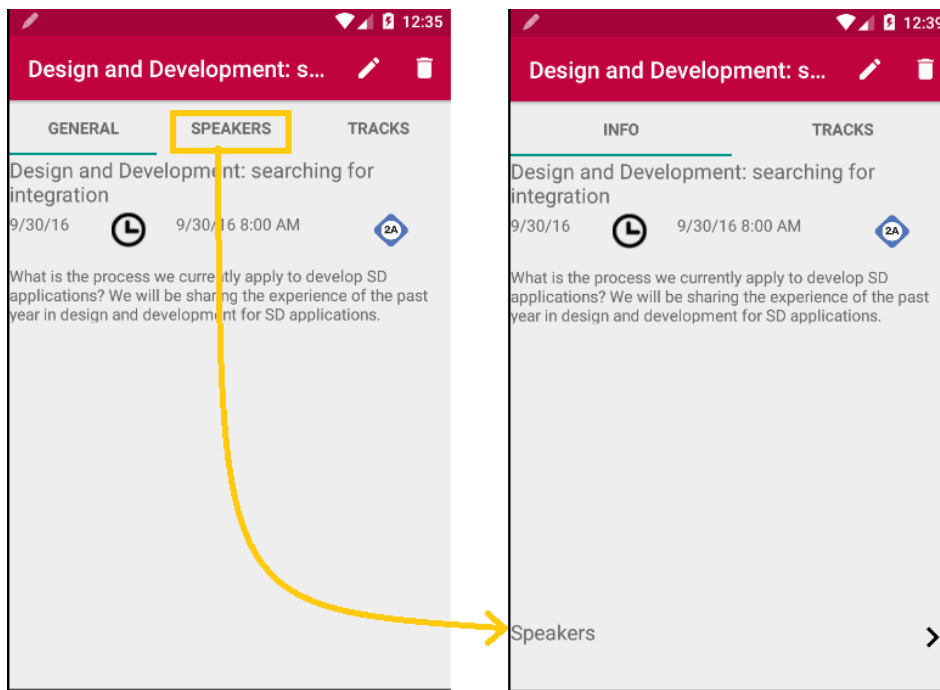


- Queremos que el grid de Speakers asociados a una conferencia salga como un grid horizontal, con hasta tres oradores por página. ¿Cómo lo implementa?



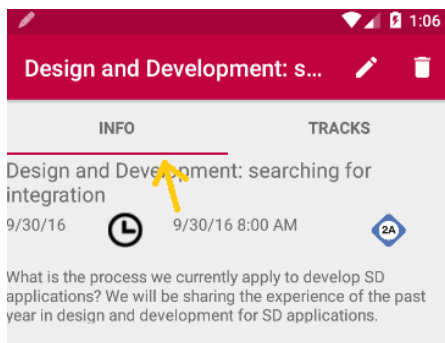
Personalizando el Detail

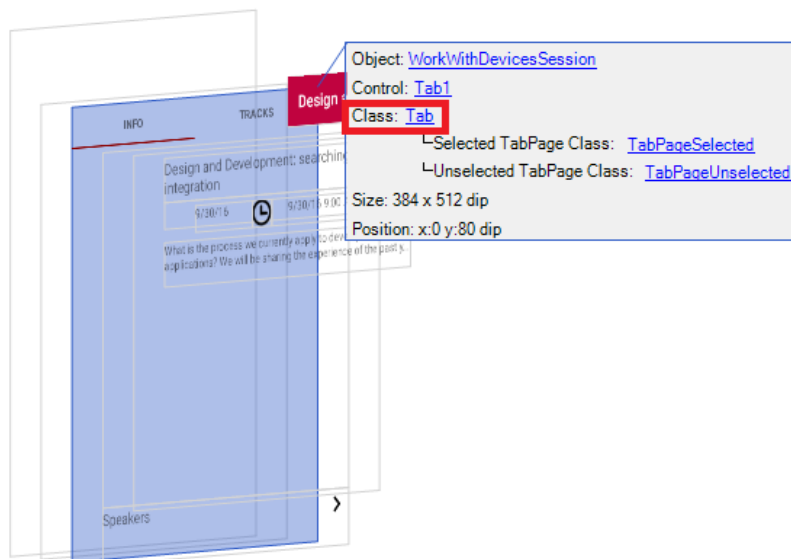
- Si en vez de mostrar la info de **Detail** como **Tabs** (imagen de la izquierda), se desea mostrar **Inline** la primera sección y como **Link** la segunda dentro de la misma tab page, y luego en otra tab page la información de los tracks, ¿cómo se consigue? (utilice Live Editing)



Recuerde que al eliminar el control <All Sections Content> del Detail, en la Toolbox reaparecerá, junto con uno por cada <Section> individual. Si coloca las dos sections individuales, recuerde observar la propiedad Rows Styles de la tabla donde quedan insertadas, y asegúrese de colocar el valor "pd" (Platform Default) para la segunda

Nota: puede cambiar el subrayado que indica que un tab está activo, yendo a la class Tab del tab control y modificando su propiedad "Tab Strip Indicator Color" (hágalo a través del Live Inspector para probar hasta encontrar el color deseado):

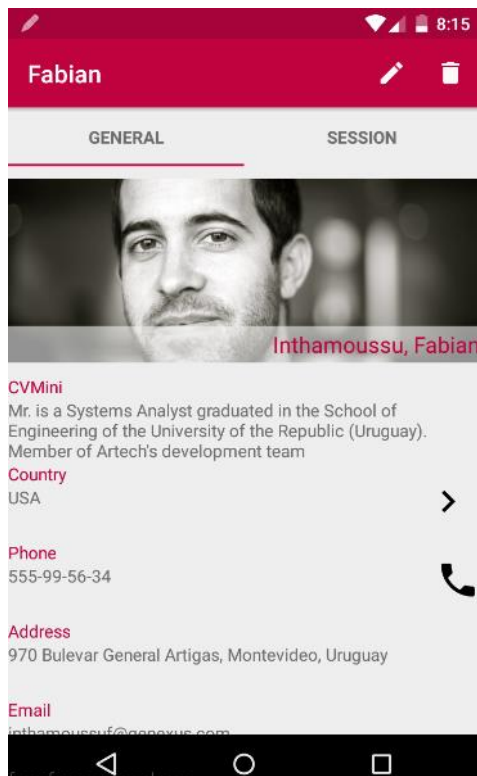




Asegúrese de guardar los cambios para que tomen efecto la próxima vez que se compile la aplicación (dado que con Live Editing prueba los efectos de los cambios, pero hasta que no salva, no se incorporan).

Personalizando las Pantallas de View y Edit

- Personalizar la pantalla de **View** del **Detail** del Speaker para que luzca en ejecución como se muestra en la siguiente figura (utilizar Live Editing para ir viendo los cambios en el emulador o dispositivo a medida que va modificando el View):





Para ello:

Deberá utilizar el control Canvas, para superponer foto y nombre completo del orador. Asegurarse que la fila donde se encuentre el Canvas ocupe 160dips.

Deberá colocar autogrow en true para el CVMMini, para que pueda crecer.

Deberá cambiarle la clase a todos los atributos que no están dentro del Canvas, para que aparezcan con el título en color (observe que cuenta con la clase AttributeBrandColored en el theme).

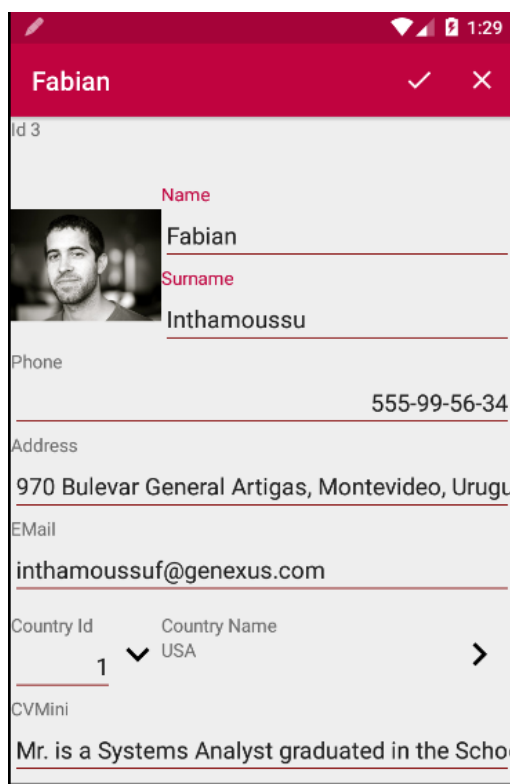
Deberá cambiarle la clase a SpeakerFullName, para que se vea en ese color y tenga un background con transparencia. Observe que cuenta con la clase AttributeTitle_BrandColorTransparentBackground en el theme.

Observe que el valor de la propiedad Height de la tabla del Canvas es "pd". Cámbielo a 160dip. Modifique la clase de la imagen por ImageFillKeepingAspectRatio y AutoGrow en False.

Posiciones absolutas para la imagen: top:10 dips; Left: 0dip; Bottom: 100%; Right: 0dip; Width: 100%, Height: 150dip; Z-Order:0.

Posiciones absolutas para el SpeakerFullName: top: 100%; Left: 0dip; Bottom: 0dip; Right: 0dip; Width: 100%; Height: 30dip; Z-order: 1. Y Horizontal Aligment = Right si quiere verlo sobre la derecha.

- Personalizar también la pantalla de Edit, para que luzca más o menos como sigue:



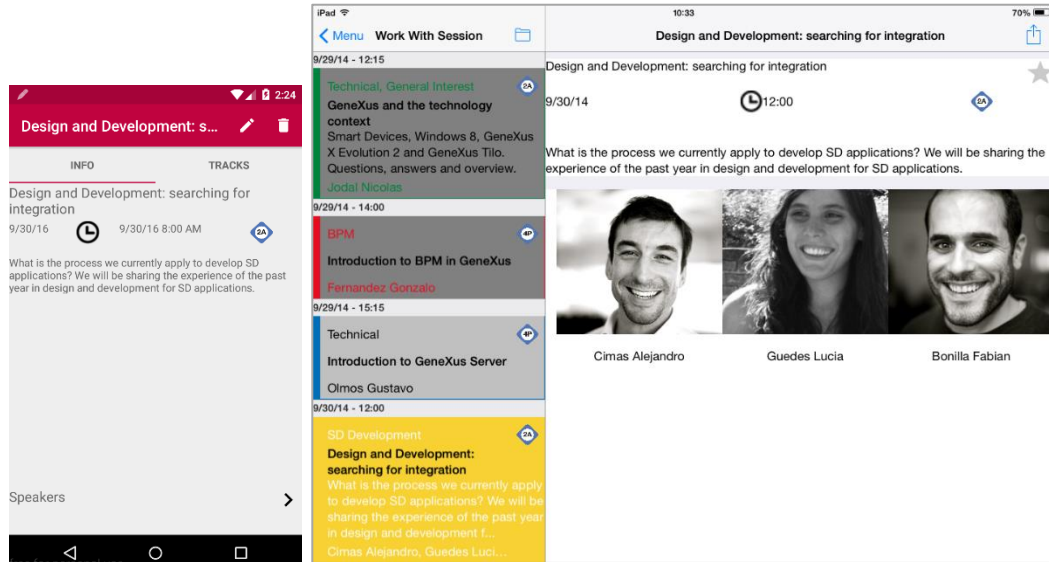
El objetivo es familiarizarse con los distintos **layouts** de un mismo objeto. Irá probando con Live Editing. Cuando el resultado sea el que deseaba, grabe y luego compile el main.



Si se desea que el **Detail** del **Work With Sessions** se muestre de manera diferente cuando se ejecuta en **Android** que cuando se ejecuta en **iOS**, tal como se ve en las imágenes que siguen, ¿cómo se implementa?

Observe que en iOS se están mostrando los speakers como un grid horizontal en la misma pantalla en la que se muestra la información general y sin mostrar los tracks, mientras que en Android los speakers se visualizan en otra pantalla,

siguiendo un link y los tracks en otra tab page. Como los cambios en el layout del Detail de Sessions los realizó anteriormente para todas las plataformas, ¿qué debería hacer para que la pantalla en iOS se vea como en la imagen de abajo?

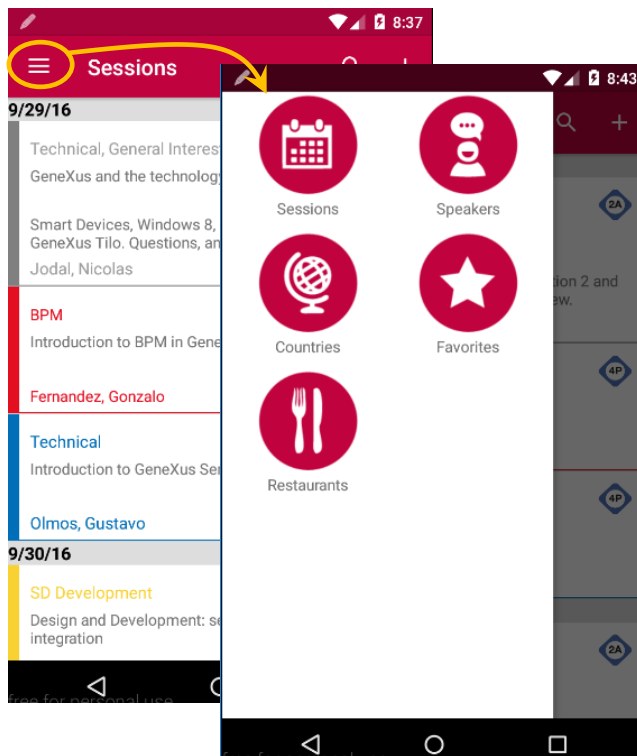


Modificando el Estilo de Navegación (Navigation Style)

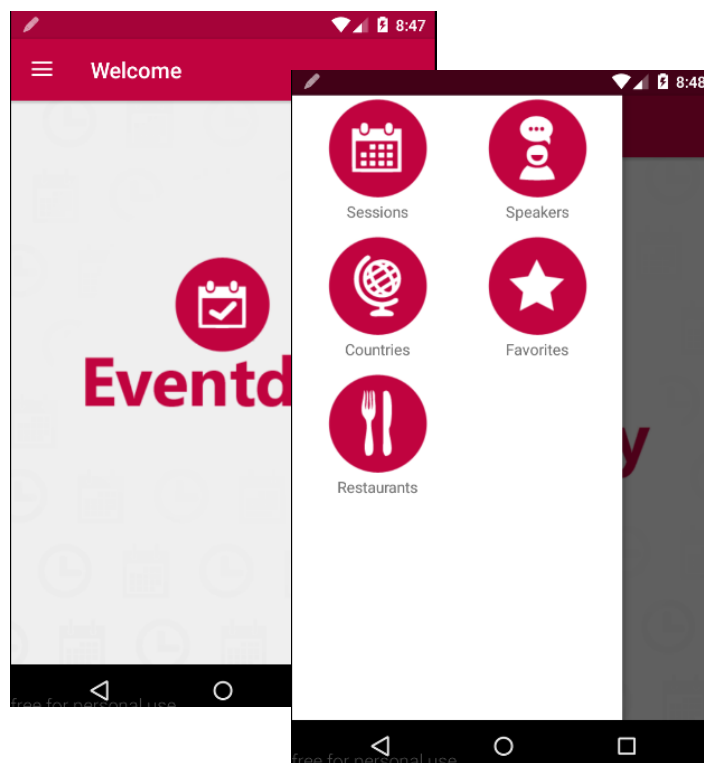
- Modificar el estilo **Navegación** de la aplicación para que al ejecutarse en teléfonos Android sea **Slide**
- Ejecutar y probar los cambios realizados



Observar que se puede acceder al Menu for Smart Devices desplazando con el dedo el borde izquierdo de la pantalla hacia la derecha, o con el menú de hamburguesa que aparece en la Application Bar, a la izquierda:



- Observar que se abre directamente el **List de Sessions**
- Si desea que en ese caso se muestre una **pantalla de bienvenida**, en lugar del List ¿cómo lo implementa? Tendrá que crear un **Panel for Smart Devices** para implementar esa pantalla. Ahora, en ejecución, cuando abra la aplicación, se verá la pantalla como se muestra en la figura de la izquierda. Y podrá acceder a todas las opciones deslizando el menú: (pantalla derecha).



- Si tiene algún dispositivo distinto de un Android Phone, pruebe en él. No deberá abrirse la pantalla de bienvenida.
- Ahora vuelva al estilo de navegación Default y observe si se ejecuta o no la pantalla de bienvenida.

Transitions

Hacer que el List de Sessions se abra desde abajo (Push up) cuando se lo invoca desde el menú y sólo en ese caso. Si todavía no estudió el tema Invocations al final del módulo de diseño y comportamiento, entonces impleméntelo en forma estática.

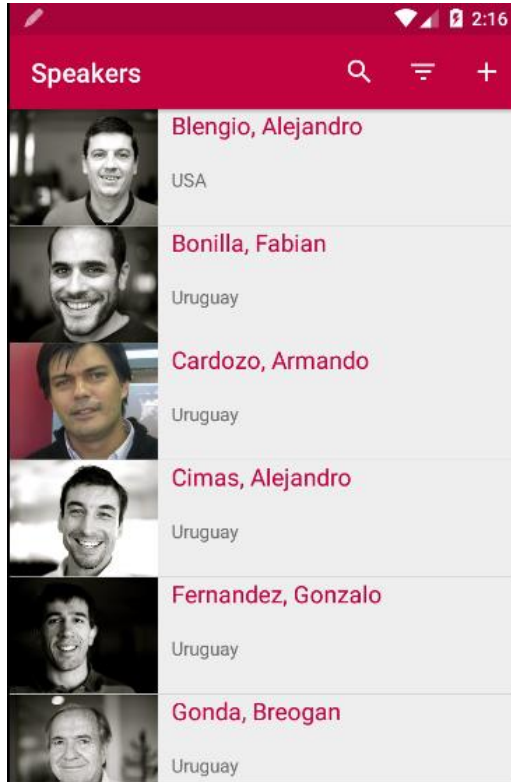


En forma estática: deberá crear una subclase de la clase Form, por ejemplo, una de nombre FormPushUp y asociársela a la MainTable del List de Sessions.

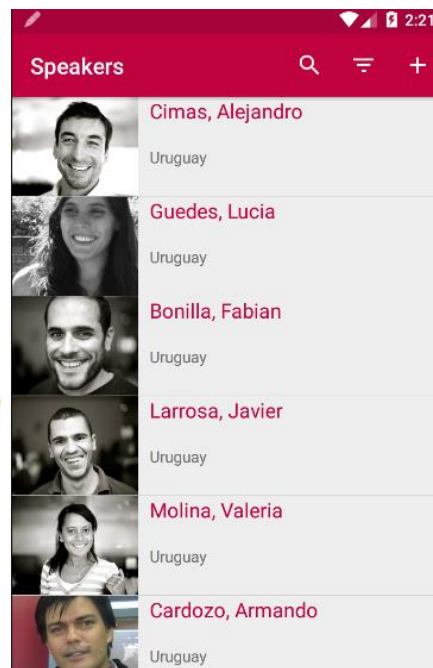
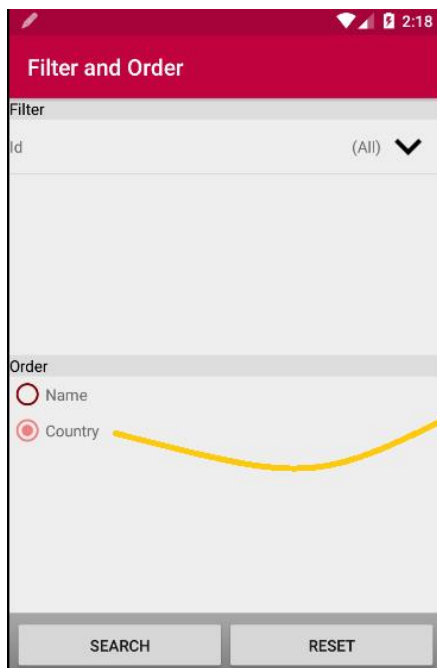
En forma dinámica: recordar las CallOptions para configurar una invocación. Recuerde la existencia del dominio enumerado “Effect” y también que para colocar más de una sentencia dentro de un evento del cliente deberá utilizar el comando Composite.

Órdenes y filtros

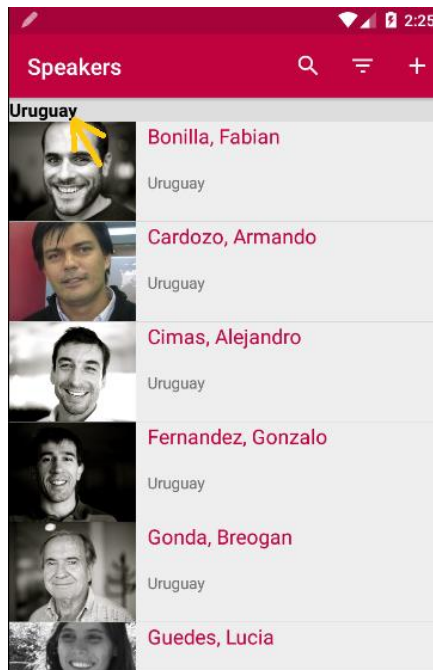
- Abra el List de Speakers y observe el orden por el que se está mostrando la información. Vaya a GeneXus y modifíquelo por SpeakerFullName. Pruébalo en ejecución.



- ¿Y si ahora quiere además dar la posibilidad al usuario de ordenar por país? Impleméntelo.



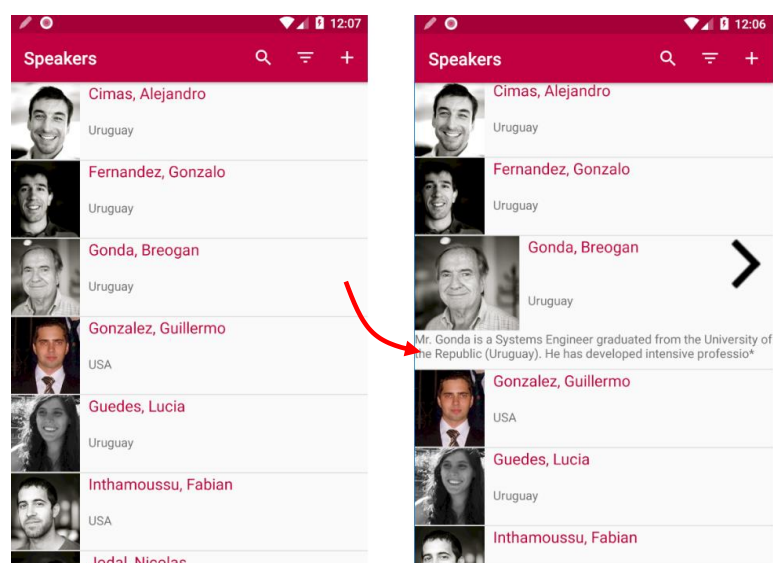
- ¿Y si quiere que de elegir esta segunda opción la información se muestre agrupada por país y ordenada dentro de cada país por SpeakerFullName?



- Observar los atributos por los que se puede hacer Search. Quitar algunos (por ejemplo el CVMini) y probar.

Comportamiento

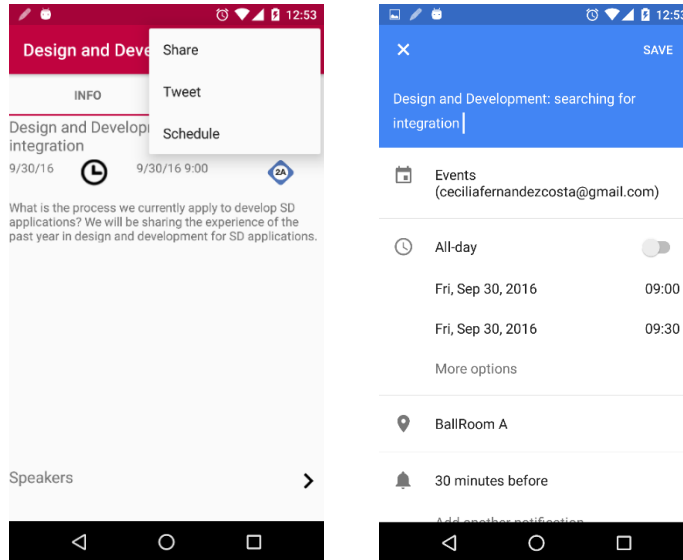
- Querremos mostrar sólo los primeros 125 caracteres del CV de un Speaker en el List y no todo su contenido y que además termine en un asterisco:



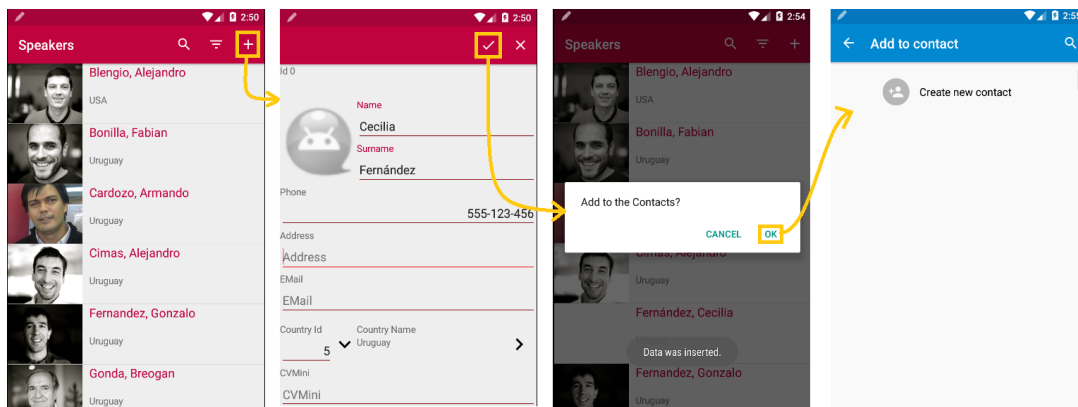
Pista: recuerde la función **substr**. También le puede ser útil el método **Length**. ¿En qué evento programará la carga de la variable con el CV reducido? ¿Se ejecuta en el Cliente o en el Server?

Integración con otras apps del dispositivo

- Observar que al visualizar la información detallada de una conferencia, se quiere ofrecer tanto compartirla “share”, escribir un tweet, o ingresarla al calendario del dispositivo.



- Agregue los tres botones y prográmelos. Probarlo con Live Editing y luego en un dispositivo Android. Observe la propiedad “Priority” que permite determinar qué controles de la Application Bar tendrán más prioridad que otros para ser mostrados en forma directa (y no en submenú). ¿Agregó los botones en el layout del Detail o en la Section(General)?
- Si ahora desea que cuando se inserta un nuevo Speaker en el sistema (a través del List), inmediatamente se ofrezca insertarlo en la libreta de direcciones del dispositivo, ¿cómo lo implementa? Puede pedirle confirmación al usuario antes de intentar insertar el contacto en la app correspondiente.



Solución: siendo &speaker una variable de tipo Business Component.

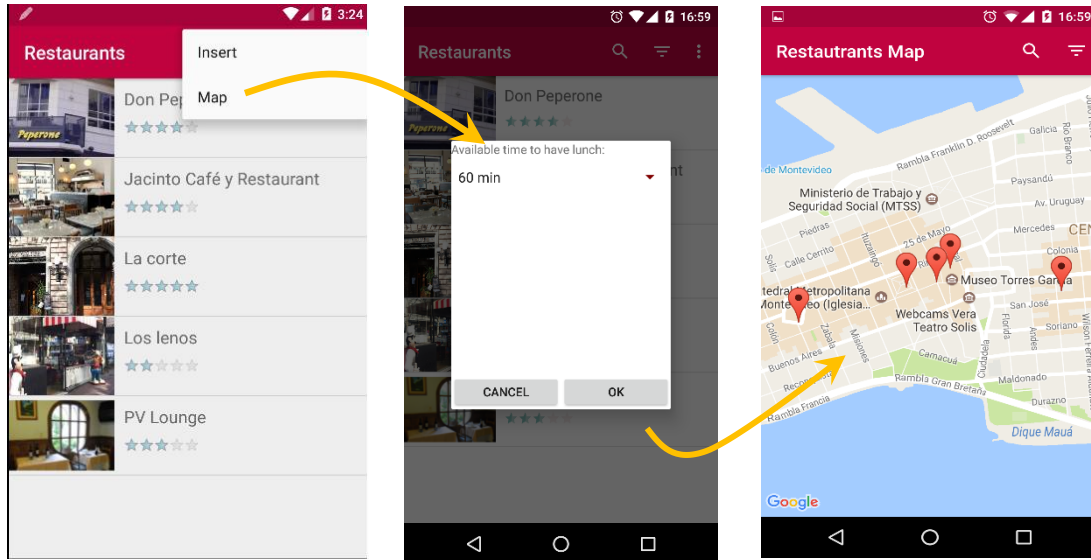


```
Event 'Insert'
Composite
    WorkWithDevicesSpeaker.Speaker.Detail.Insert(&speaker)
    Confirm( "Add to the contacts?")
    Contacts.AddContact(&speaker.SpeakerName, &speaker.SpeakerSurname,
        &speaker.SpeakerEmail, &speaker.SpeakerPhone, "", "", "")
endcomposite
EndEvent
```

Opciones de invocación: Popup

Ahora deseamos que el List de Restaurants se siga mostrando como un grid estándar, pero darle la posibilidad al usuario de filtrar los restaurantes de acuerdo al tiempo máximo del que dispone para almorzar (observar que la transacción Restaurant tiene el atributo RestaurantQuickestServiceTime del dominio enumerado que se importó con un xpz), y mostrar los restaurantes resultantes en un mapa, ¿cómo lo implementa? Queremos que la pantalla que pide al usuario ese tiempo máximo, sea una ventana Popup.

Puede crear los paneles de cero, o importarse el archivo 4-Restaurants.xpz.



Solución: agregar botón Map al List de Restaurants, y variables &timing y &ok.

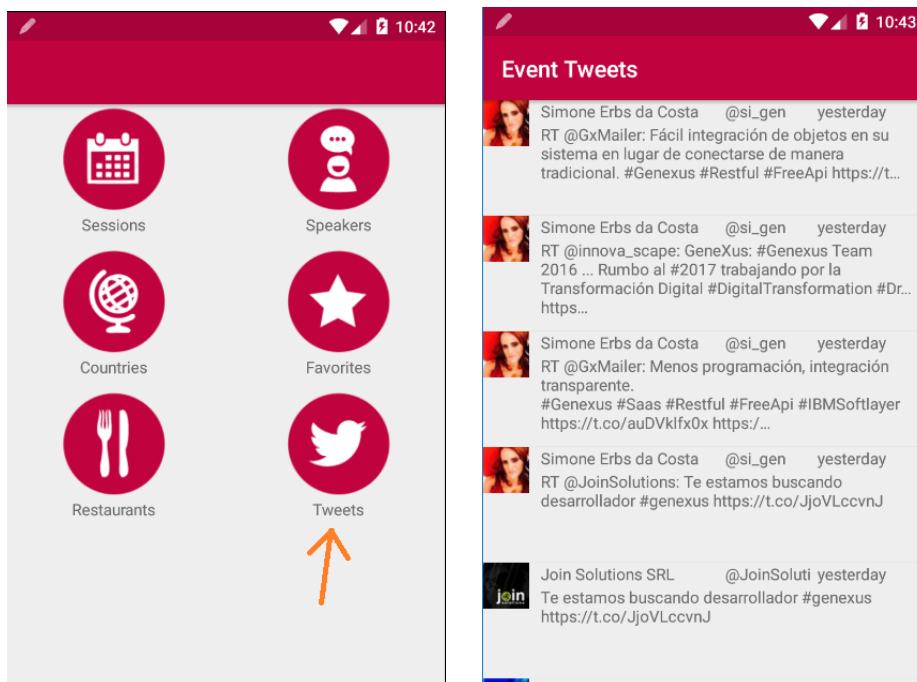
```

Event 'Map'
  Composite
    RestaurantFilter.CallOptions.Type = CallType.Popup
    RestaurantFilter.CallOptions.TargetSize = CallTargetSize.Small
    RestaurantFilter( &timing, &ok )
    If &ok
      RestaurantsMap(&timing)
    endif
  endcomposite
Endevent
  
```

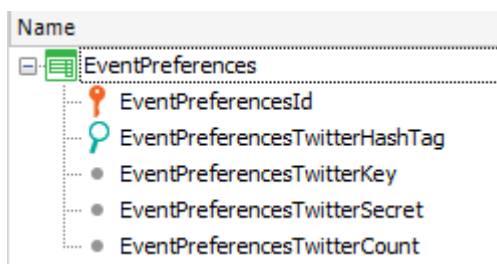
Además obsérvese que deberá asociarle conditions al grid de Restaurants Map para que se filtren los restaurantes de acuerdo al tiempo recibido por parámetro.

Paneles para ver los Tweets del hashtag del Evento

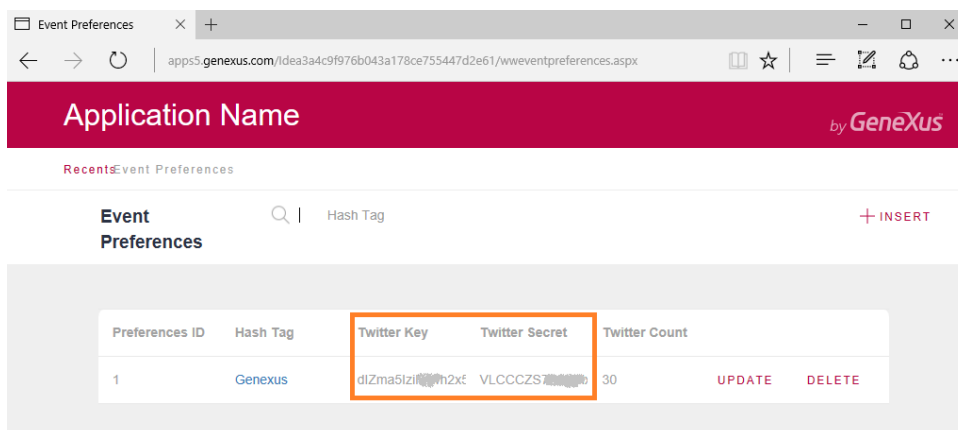
- Importe el archivo 5-EventDay-Tweets.xpz, para observar cómo se puede implementar la integración con Twitter a través de un panel para mostrar los tweets asociados al evento. Vaya a su Menu for Smart Devices y agregue una opción que invoque al Panel for Smart Devices EventTweets (cuenta con la imagen en la KB de nombre DashboardOption_Twitter que fue importada en xpz al principio). F5.



- Para que ese panel le muestre los tweets, debe configurar las credenciales de Twitter dentro de la transacción EventPreferences:

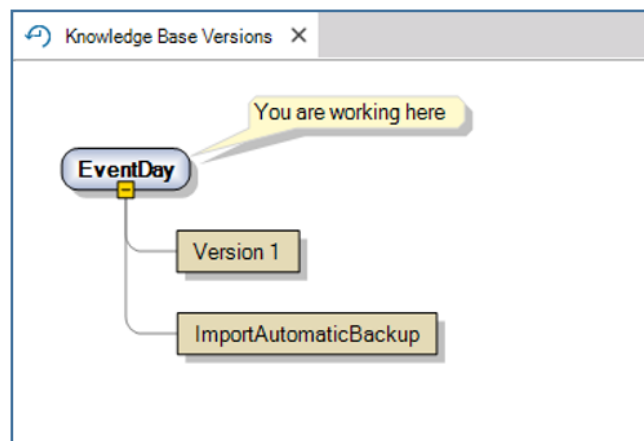


donde se encuentra el hashtag que se utilizará en Twitter para referenciar al evento, y la clave de desarrollador en <https://dev.twitter.com/>, junto con el secret que se le brinda. Pídale al instructor estos datos para probar en ejecución, yendo al Home web (ejecútelo a través de View/Show QR Codes), y allí modificando el registro 1 que ya estaba creado, pero con los datos vacíos de TwitterKey y TwitterSecret:



6. Activación del GAM

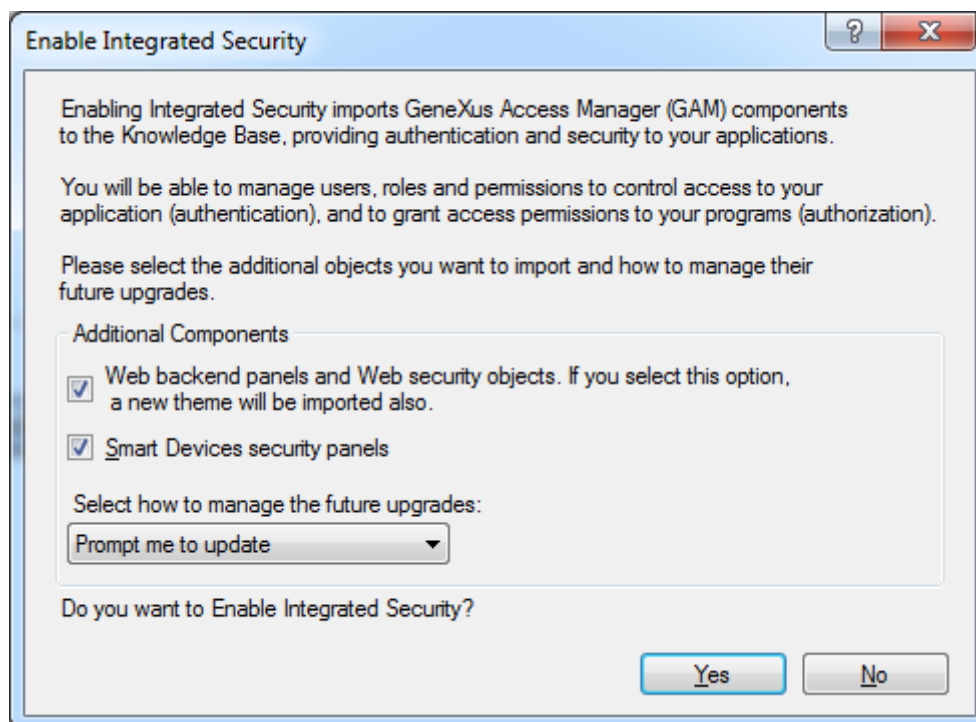
- Antes de continuar, **congele** la versión actual de desarrollo, para tener un respaldo del estado actual de la KB.
- Esto se logra desde la opción **View/Versions, Freeze** sobre la versión.



- Incorporar el **GeneXus Access Manager (GAM)** para obtener un control de acceso automático.



Para ello: Configurar la propiedad a nivel de la Versión **Enable Integrated Security** en True, y presionar Install:



Nota importante: asegúrese, cuando finalice la importación de todos los objetos del GAM, que la propiedad “Enable Integrated Security” haya quedado efectivamente en True. Si no es así, póngala nuevamente en True y presione Install nuevamente.

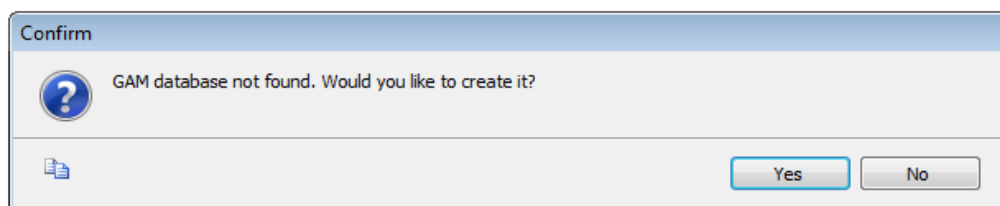


Observar:

- Las propiedades de la versión que aparecen luego de habilitar la seguridad integrada.
- Los Data Stores. Verá uno distinto para la base de datos de GAM.
- Todos los objetos que se importan en la KB tras el cambio anterior (en el KB Explorer encontrará, bajo el Root Module, dos nuevos módulos: GAM_Examples y GAM_Library)
- ¿Qué objetos son los que se configuran automáticamente en las propiedades “Login Object for Web” y “Login Object for SD”?

Enable Integrated Security	True
▼ Integrated Security	
Integrated Security Level	Authentication
Application ID	5861790c-f844-4b4c-aa53-7fb9d89831c1
▼ Web specific	
Login Object for Web	GAMExampleLogin
Not Authorized Object for Web	GAMExampleNotAuthorized
▼ SmartDevices specific	
Login Object for SD	GAMSDLogin
Not Authorized Object for SD	GAMSDNotAuthorized
Change Password Object for SD	GAMSDChangePassword

- Hacer **Rebuild All** (tomará su tiempo) y luego **F5**. Confirme que desea crear la base de datos del GAM:

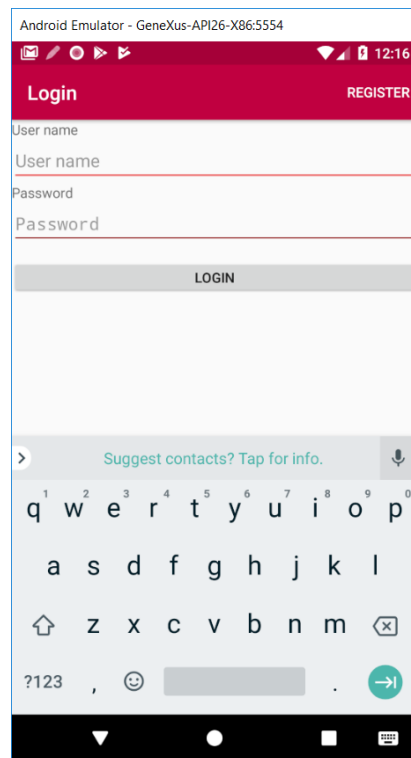


Nota: Se requiere el driver MySQL para .NET (libmysql.dll) para poder crear la base de datos del GAM contra MySQL. Si no está prototipando con .NET (en ese caso ya la tiene), asegúrese de instalarla en C:\Windows\SysWOW64 si su CPU es de 64 bits y en C:\Windows\System si es de 32 bits, desde el siguiente link: <http://sourceforge.net/projects/mysql/drivers>. Ver más información en el wiki: <http://wiki.genexus.com/commwiki/servlet/wiki?2041>

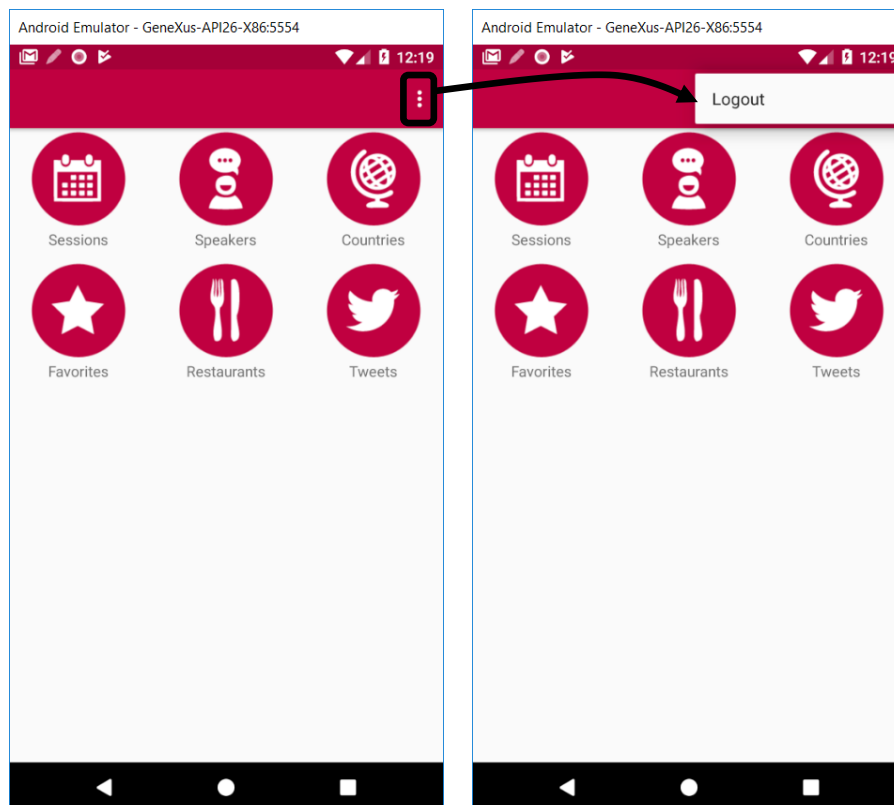
- Ejecutar la aplicación en el Smart Device o emulador. Verá que como pantalla inicial ahora aparece la de login. Ingresar con los siguientes datos de login:

User: **admin**

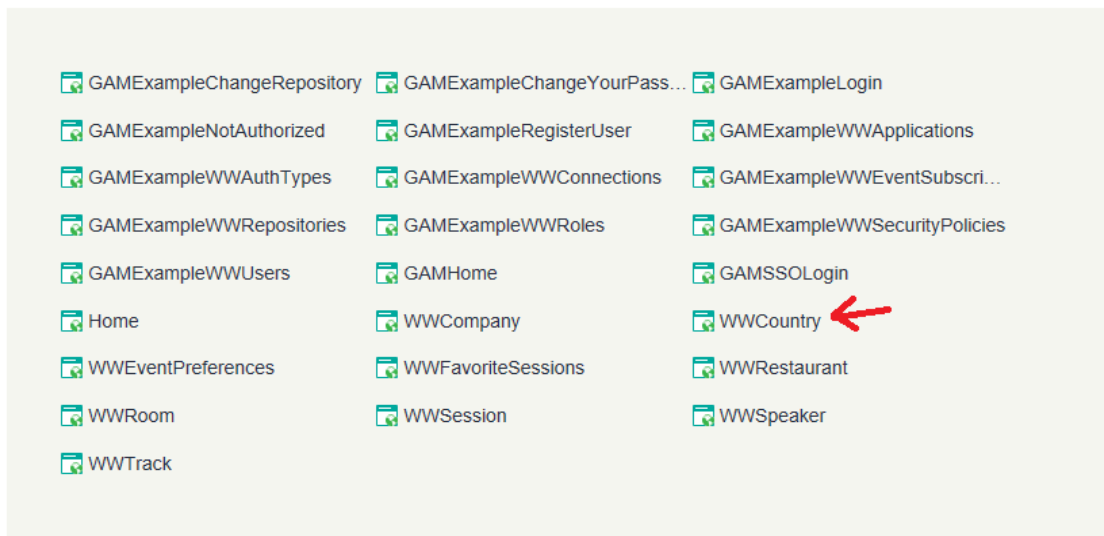
Password: **admin123**



- Observar el botón de Logout que se habrá agregado al Menú:



- Ejecutar el Developer Menu (web) —modificando el startup object por “none”, haciendo File/Build Developer Menu y luego View/Show QR Codes—, e intentar acceder a un Work with (por ejemplo, el WWCountry). No podrá sin pasar antes por el **Login**. Pruébalo.



- Ingresar con los siguientes datos:

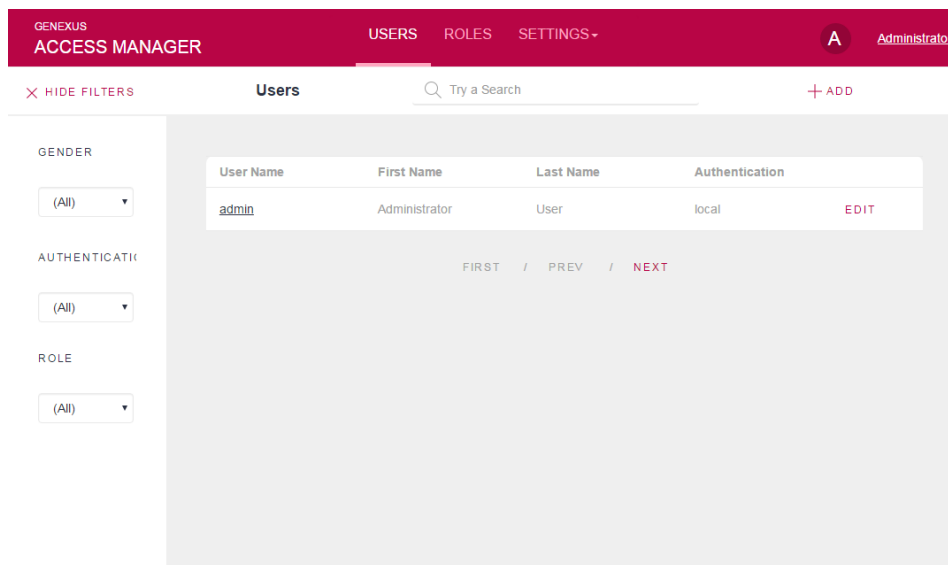
The image shows a login form with the following elements:

- A yellow error message at the top: "User must be authenticated. (GAM104)".
- The title "LOGIN" in large, bold, black letters.
- A link "DON'T HAVE AN ACCOUNT? REGISTER" in red text.
- A text input field labeled "User Name".
- A text input field labeled "Password".
- Two checkboxes: "Keep me logged in" and "Remember Me".
- A red "LOG IN" button.
- A link "FORGOT YOUR PASSWORD?" in red text at the bottom.

usuario: **admin**

password: **admin123**

- Desde el Developer Menu ejecutar el Web panel **GAMHome** y observar el Backend para administrar la **Seguridad**

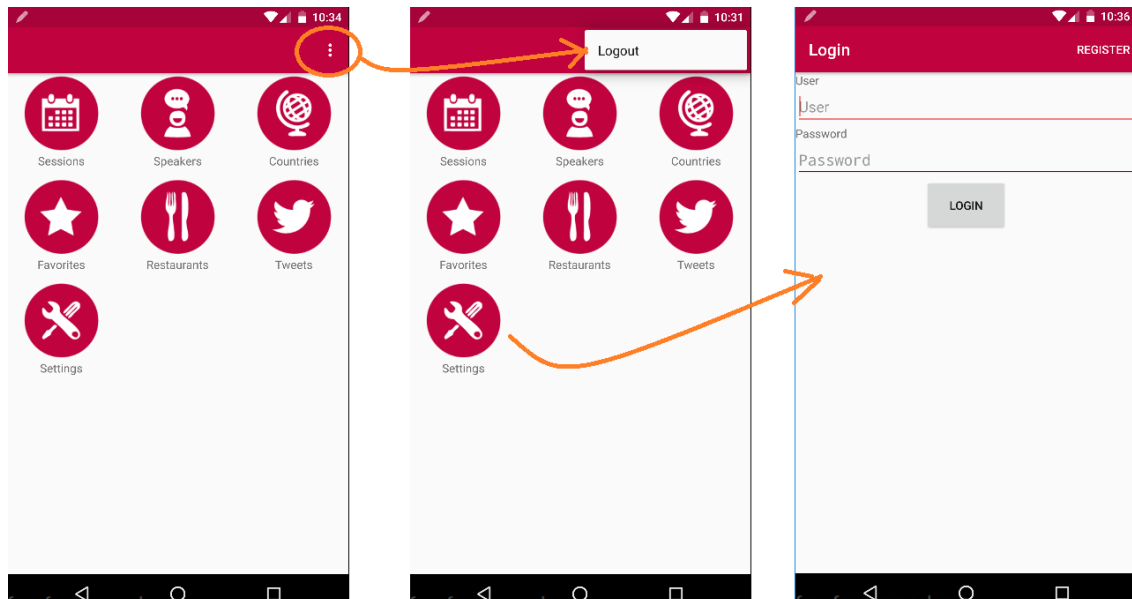


- Definir un nuevo usuario desde la opción **Users**

Asignarle rol de administrador.

Configuración de la Seguridad a nivel de Objeto

Queremos que desde la app móvil pueda trabajarse con los settings del evento. Pero sólo si se trata de un usuario autorizado. Para el resto de la app, no queremos seguridad. Implementélo.



Para ello:

- Configurar la propiedad **Integrated Security Level = None** a nivel de la Versión:

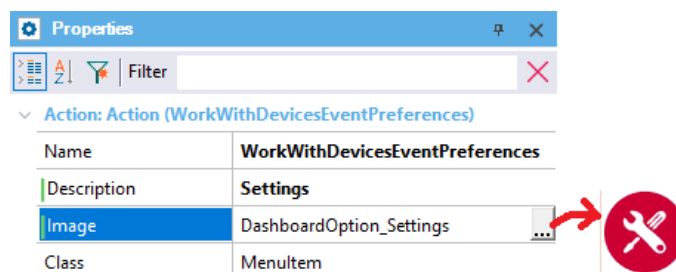
Enable Integrated Security	True
▼ Integrated Security	
Integrated Security Level	None
Application ID	5861790c-f844-4b4c-aa53-7fb9d89831c1

- Aplicar el patrón Work With for Smart Devices a la transacción EventPreferences y en el objeto WorkWithDevicesEventPreferences configurar la propiedad **Integrated Security Level** en el valor **Authentication**.



Esta configuración permite que no sea necesario loguearse para ingresar a las opciones del Evento, SALVO para ingresar a la opción Settings, donde aparecerá la pantalla de **login**.

- Agregar una acción al Menu for Smart Devices que invoque a este Work With:



- Realizar un **Rebuild All**, volver a poner al objeto EventDay como StartUp Object y ejecutar la aplicación para verificar los cambios realizados.

Nota: Al ingresar al Menu for Smart Devices, primero seleccionar **Menu/Logout** para desloguear el usuario que se había usado anteriormente. Si tiene el estilo de navegación Slide, cámbielo al Default y pruebe.

7. Convirtiendo la Aplicación a Offline

- Queremos que toda la aplicación, a excepción de la parte que requiere autenticación (en nuestro caso, los settings del evento) y el acceso a los tweets pueda ejecutarse offline. Implementelo.



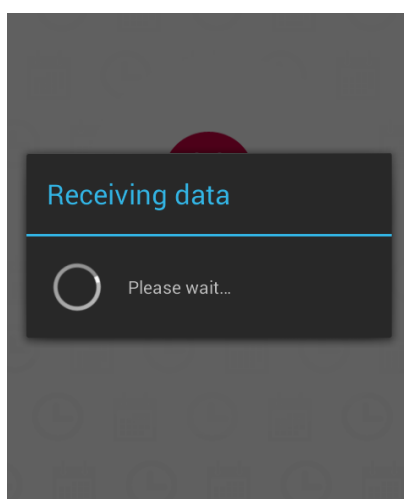
Para ello:

Configurar en el **Menu for Smart Devices** la propiedad **Connectivity Support = Offline**
Configurar en la Transacción **EventPreferences** la **Connectivity Support = Online**
Y también hacerlo en el panel EventTweets.



Recordar que el login del GAM solo se puede realizar desde un objeto Online.

- Hacer un **Rebuild All** y verificar que se crea el objeto **Offline Database** indicando en el **Smart Devices Impact Analysis** cuáles serán las tablas que se crearán en el dispositivo.
- Configurar el mecanismo de **Sincronización** de datos (Propiedades para el Send y Receive) que desee. Es decir, cuándo desea enviar los datos (en nuestro caso, los que se puedan haber ingresado/actualizado/eliminado en el dispositivo a través de los Work With que tienen los modos de Edit disponibles) y cuándo y cómo recibir datos del servidor cuando se esté conectado.
- Ejecutar la aplicación (recuerde que deberá necesariamente ejecutarla compilada. No podrá utilizar el KBN) y observar que se realiza la carga de datos en la **base de datos local** del dispositivo.



- Desconectar** el dispositivo y verificar que la aplicación funciona correctamente. Ingrese algún registro nuevo, por ejemplo, un nuevo país.
- Intentar ingresar a la opción **Settings**, verificar que el **login** sólo funciona si el dispositivo está **conectado**.

- Intente ver los tweets. Se le desplegará un mensaje: “No Internet Connection”.
- Ingrese algún país nuevo a través del Developer Menu web. Observe que el país que había ingresado antes desde el dispositivo no está en la DB centralizada y que, a la vez, el país que acaba de ingresar en la DB centraliza a través de la app web, no se muestra en el List de países del dispositivo.
- Conecte el dispositivo a internet y observe que los datos son sincronizados, de acuerdo a los criterios que usted definió para send y receive (¿dejó para receive el criterio “On launch application” y after 600 seconds? ¿Qué deberá hacer entonces para recibir en el dispositivo los cambios hechos a través del Developer Menu?)
- Pruebe cambiar la propiedad “Data Receive Criteria a After Elapsed Time”, ingrese un país a través del Developer Menu, refresque el List de países en su dispositivo inmediatamente, espere la cantidad de segundos que tiene indicada en la propiedad “Minimum Time Between Receives” y vuelva a refrescar.