

Controles: etiquetas, tablas e imágenes. Sus particularidades en el diseño.



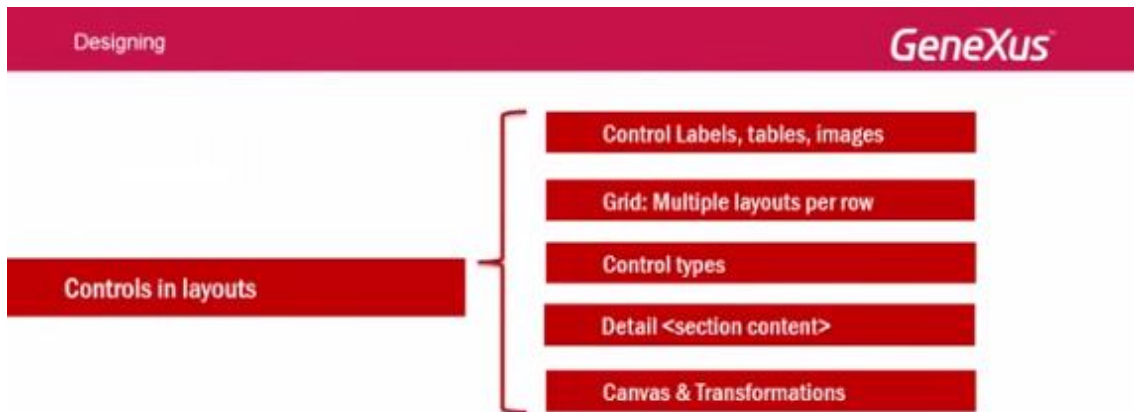


Developing the mobile application

Designing: controls in layouts

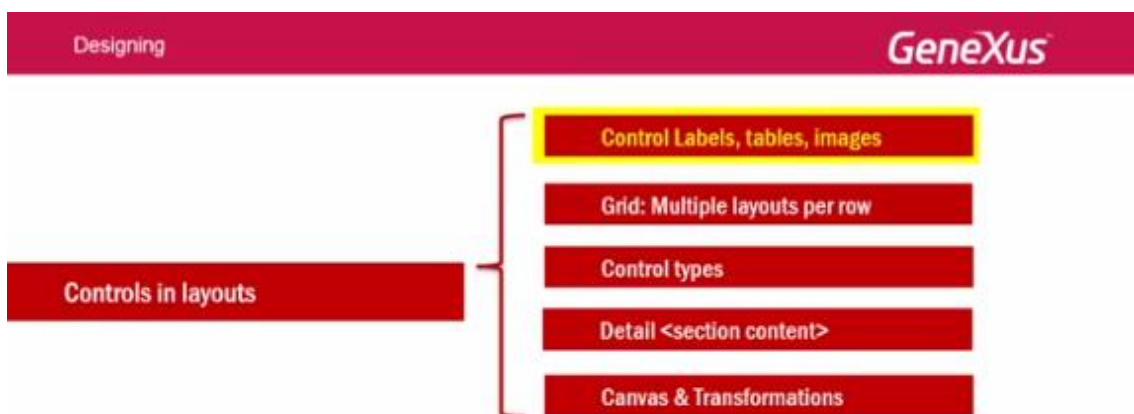
Labels, tables, images

Cecilia Fernández | GeneXus Training

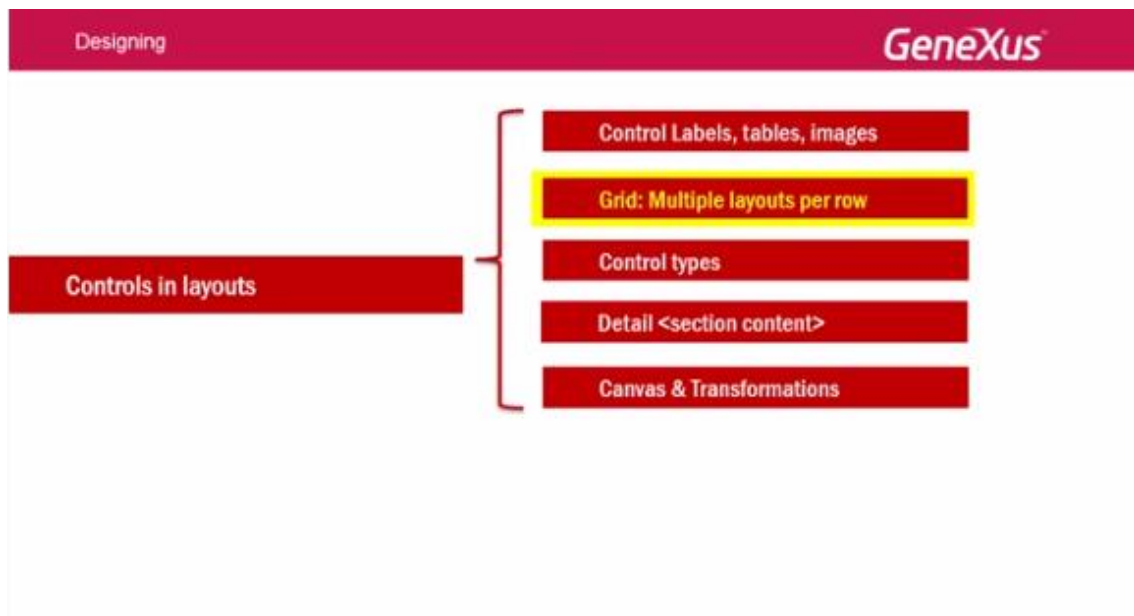


En el conjunto de videos que siguen, nos dedicaremos a las particularidades que asumen los controles en los Layouts respecto al uso conocido.

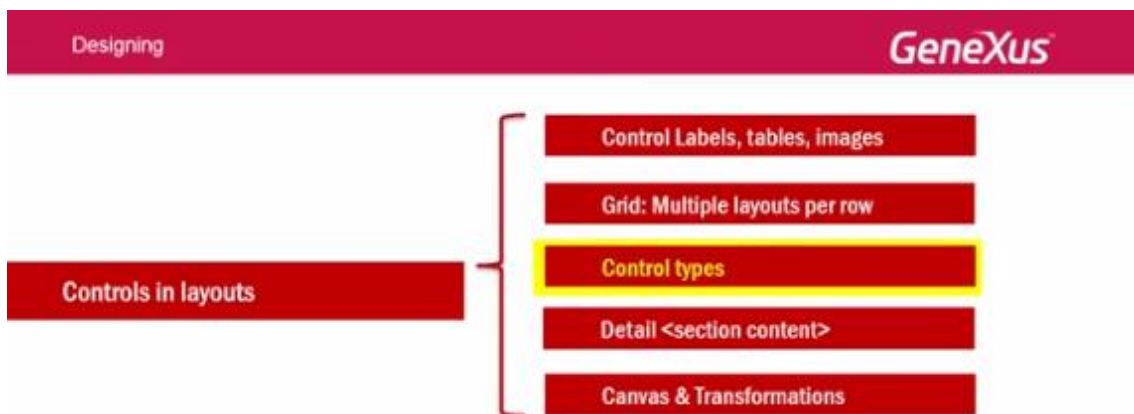
Fundamentalmente en 5 aspectos:



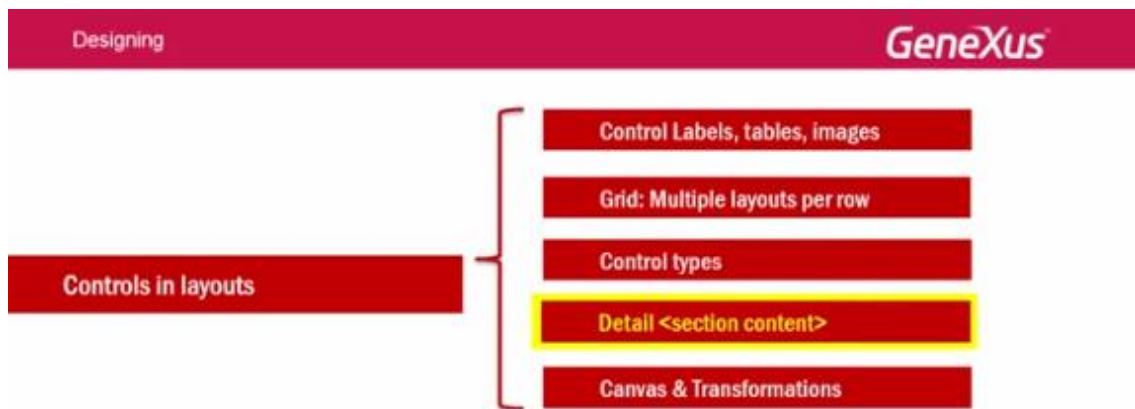
Las particularidades de los controles variable/atributo en lo que hace a las etiquetas, el uso de las tablas y las particularidades del control imagen.



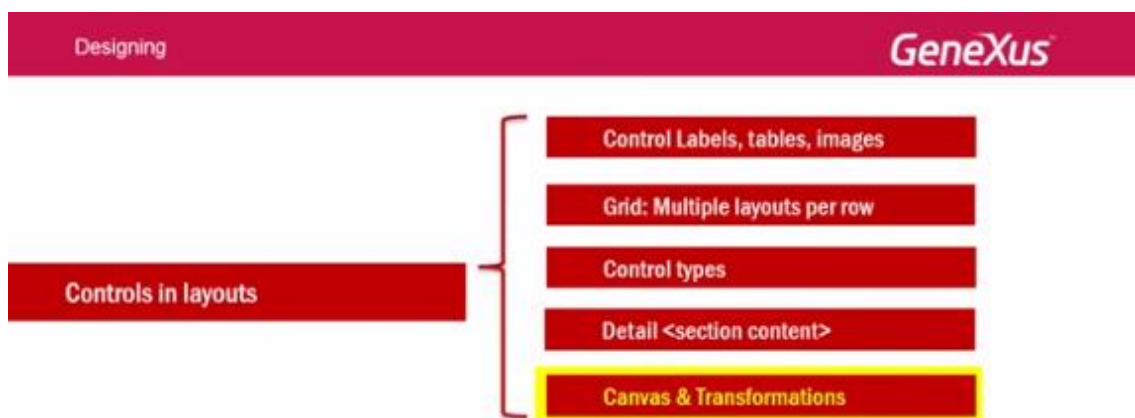
Las particularidades que presentan los grids, respecto al diseño y comportamiento de cada línea.



La posibilidad de modificar para un control su tipo, para que luzca y se comporte de una manera diferente a la predefinida.



Los contenedores de secciones para el Detail de un Work With

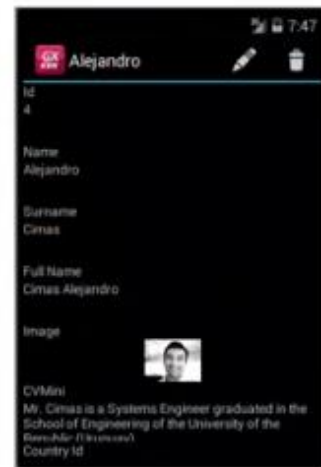


Y por último, la posibilidad de definir un control que asuma posicionamiento absoluto, pudiendo solaparse a otro en el Layout; permitiendo pensar en capas y por tanto en profundidad en el eje espacial. Además veremos esta funcionalidad en combinación con otra relativa al comportamiento: la posibilidad de desplazar, resizear, rotar, escalar un control por la pantalla (estas son las transformaciones).

Empecemos por el primer punto:

Controls in layouts

2 areas



Dos áreas componen un layout.

El área conocida como Application Bar:

Controls in layouts

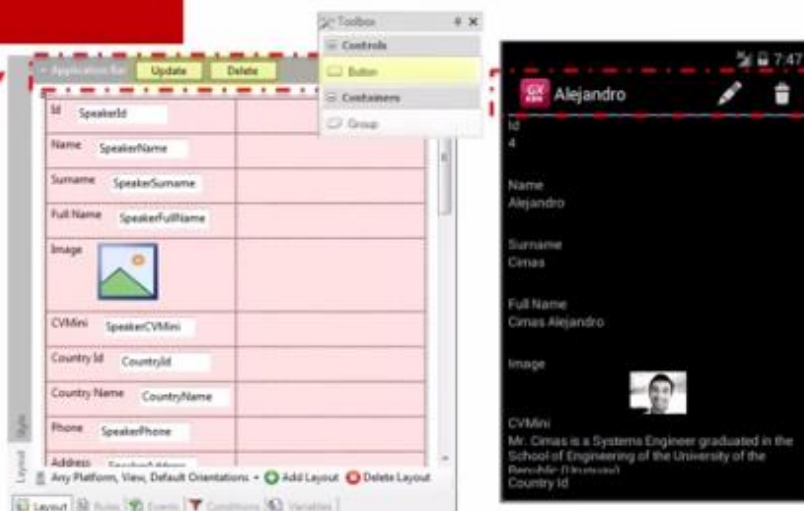
2 areas



Donde se pueden colocar botones

Controls in layouts

2 areas

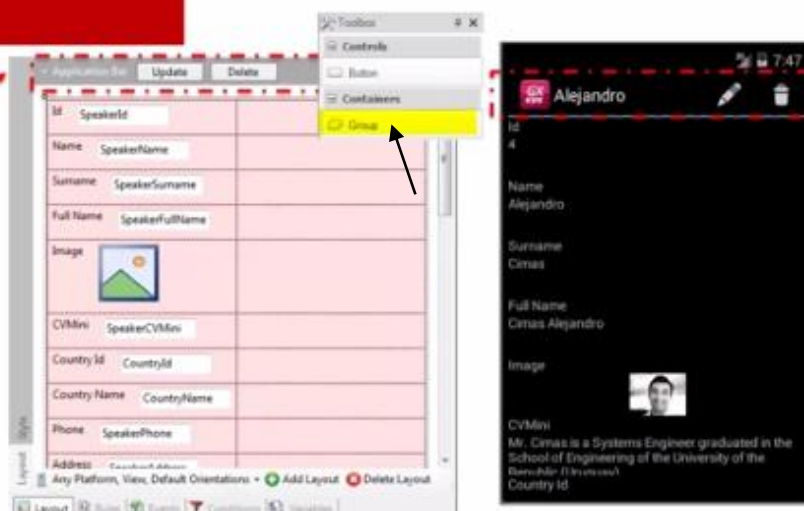


Que tengan asociadas imágenes o no.. que corresponderán a acciones que se quieren realizar.

O contenedores de grupos de acciones: Group Control

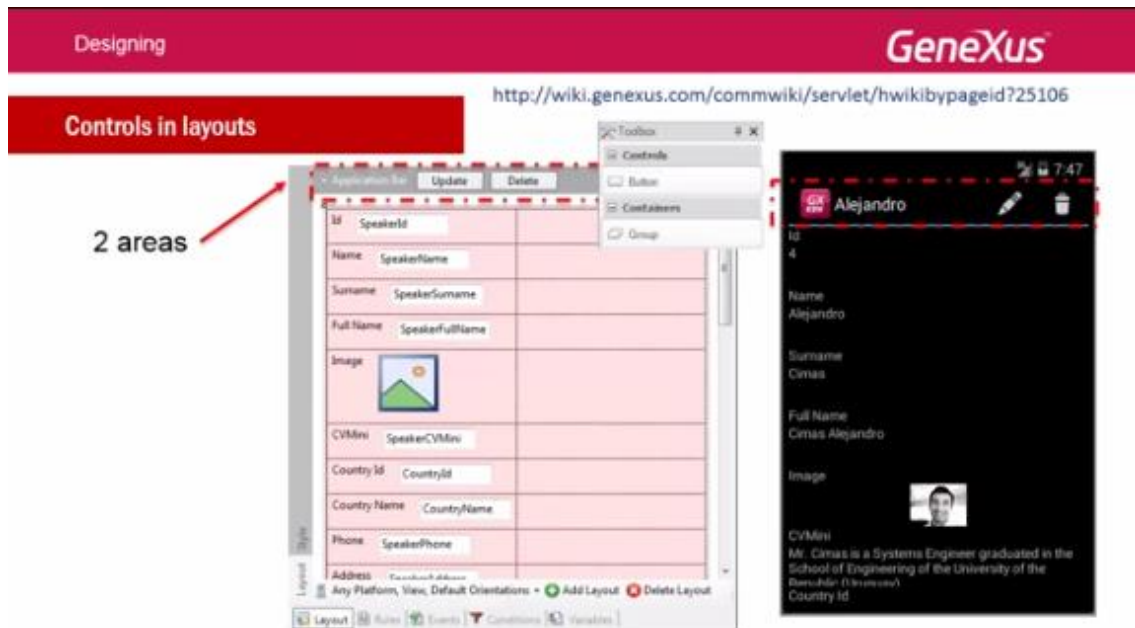
Controls in layouts

2 areas

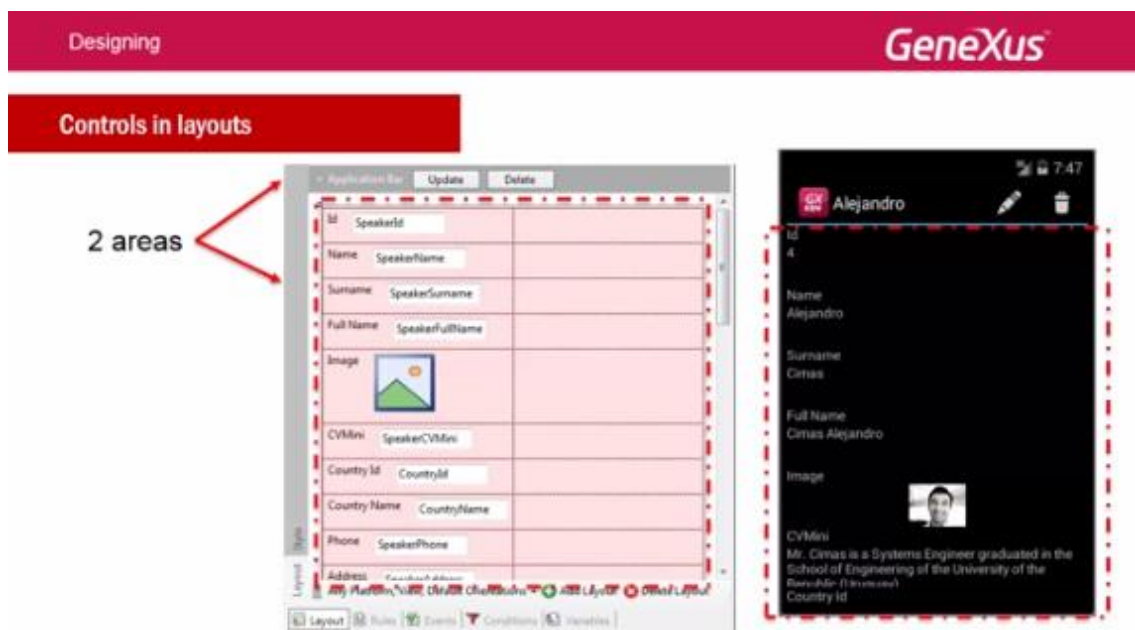


Para agrupar varias acciones y ofrecerlas por ejemplo, como menús desplegables.

Podrá ver más información sobre este tam, aquí:



La segunda área, es el área del Layout propiamente dicha:



que nos ofrece la Toolbox:

Controls in layouts

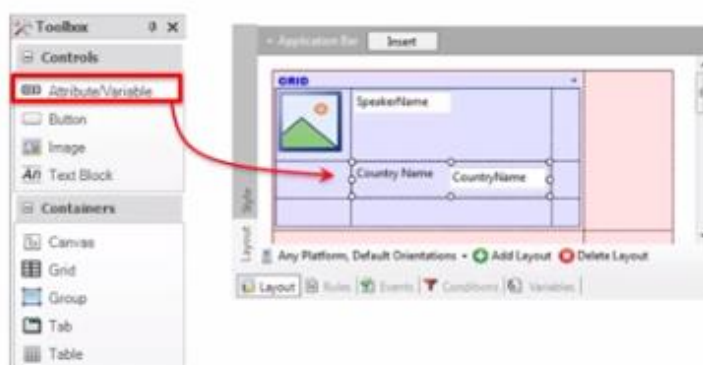


Para poder insertar los controles.

Si se trata del nodo Detail del Work With, aparecerán además, Place Holders para las secciones, como veremos más adelante.

Nos concentraremos en esta segunda área y dejaremos la primera para más adelante cuando estudiemos los eventos.

Controls in layouts: atts/vars



Para los controles de atributos o variables, tenemos la **propiedad** Label Position:

Controls in layouts: atts/vars



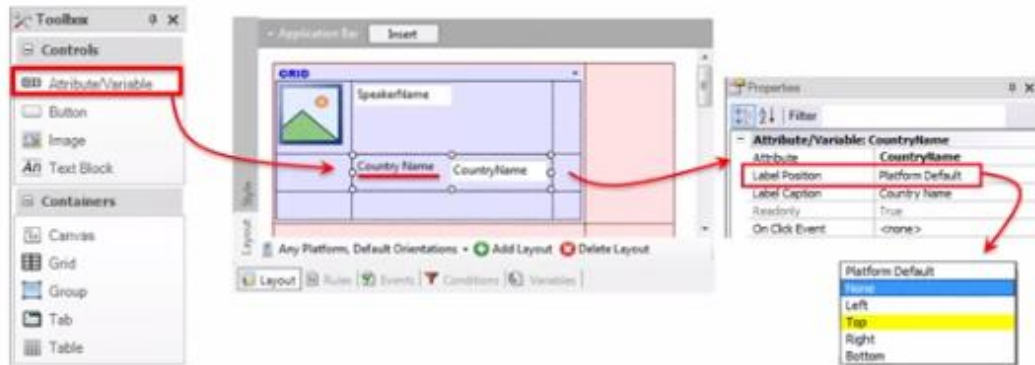
Que permitirá especificar la posición que ocupará la etiqueta del atributo o variable en cuestión:

Controls in layouts: atts/vars



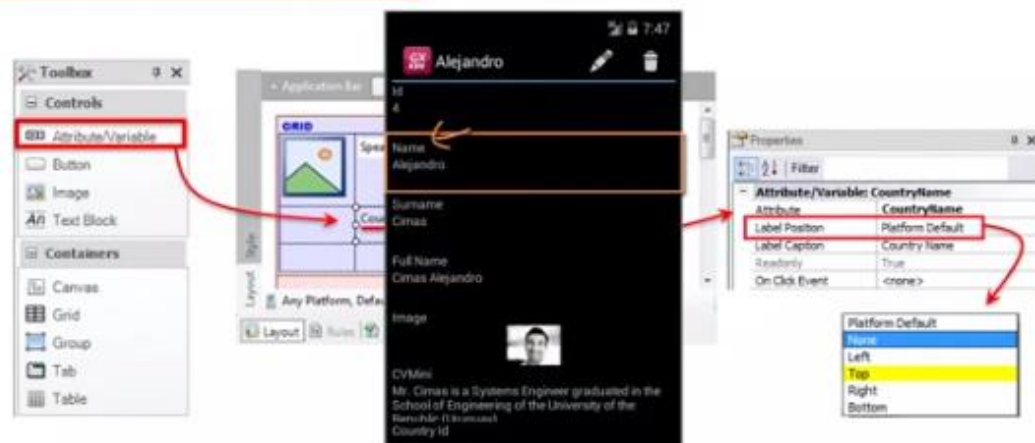
Cada plataforma tiene su default. En Android por ejemplo, es: Top

Controls in layouts: atts/vars



Lo que significa que la etiqueta aparecerá arriba del control atributo / variable:

Controls in layouts: atts/vars

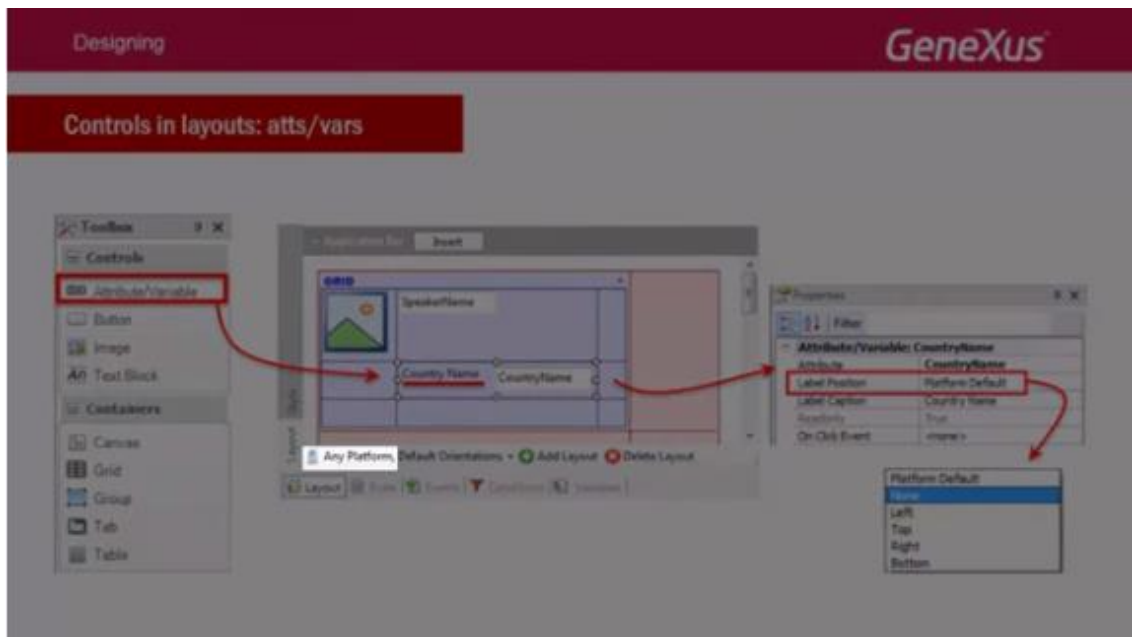


No como lo estamos viendo aquí, que es a la izquierda: Left

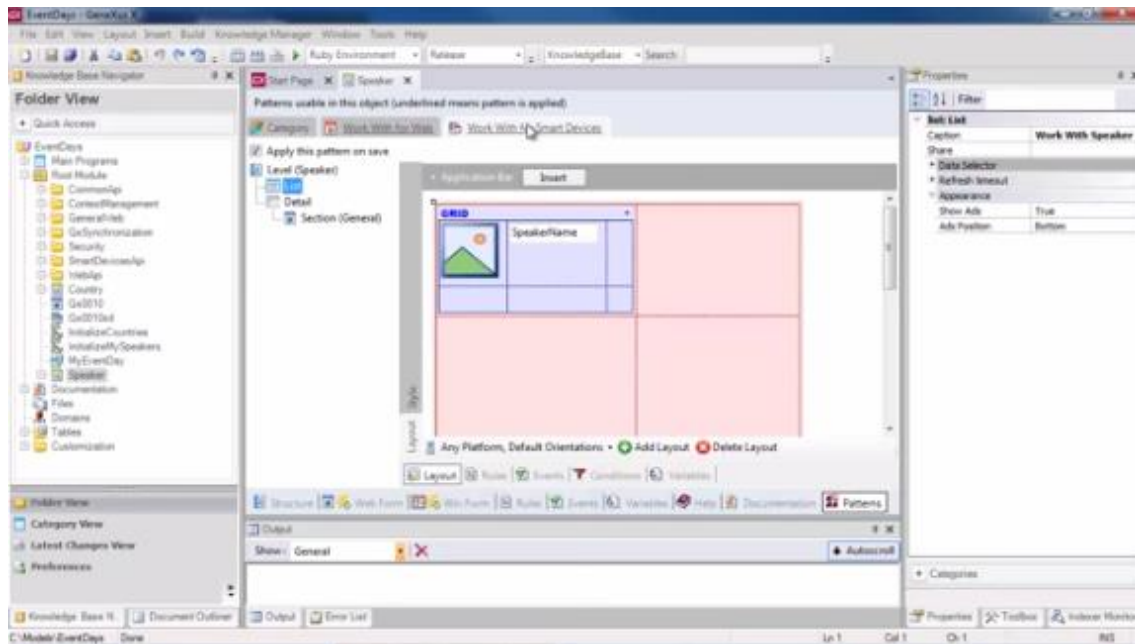
Controls in layouts: atts/vars



Porque de algún modo hay que mostrarlo, y esta pantalla es genérica, independiente de la plataforma



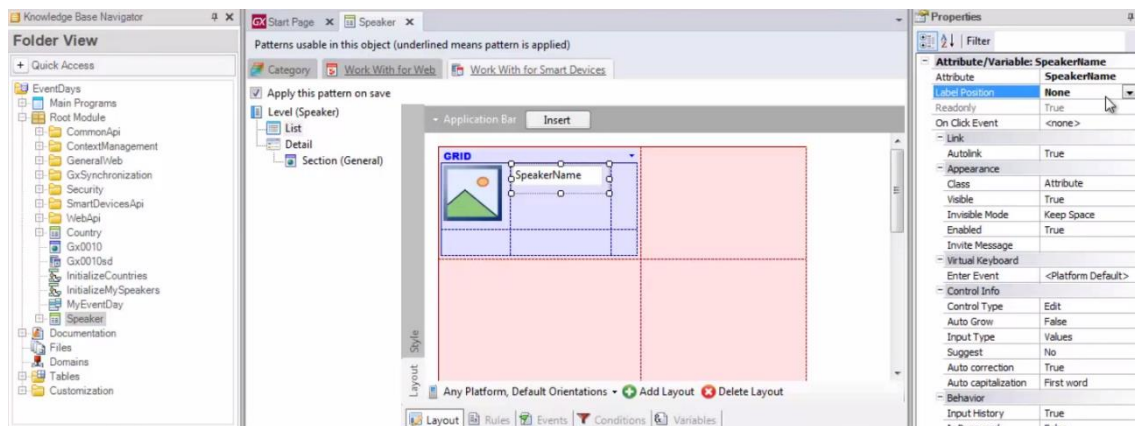
Veámoslo en GeneXus:



Estamos en el List del Work With for Smart Devices de la transacción Speaker.

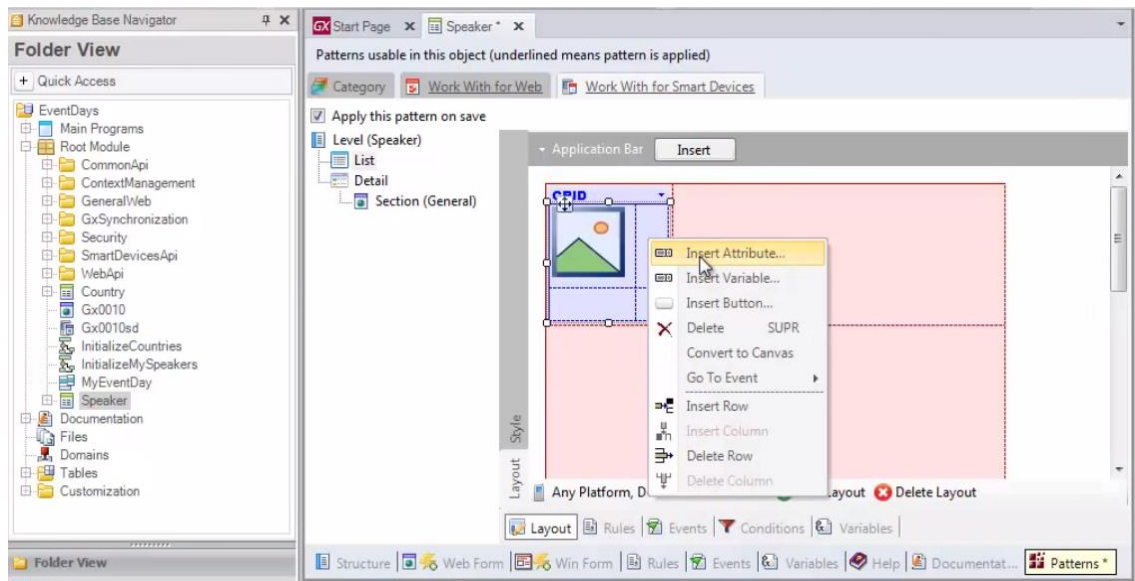
Observemos que el pattern, automáticamente colocó el control atributo **SpeakerName** en el grid.

Si observamos sus propiedades, vemos que la **Label Position** asume el valor: None

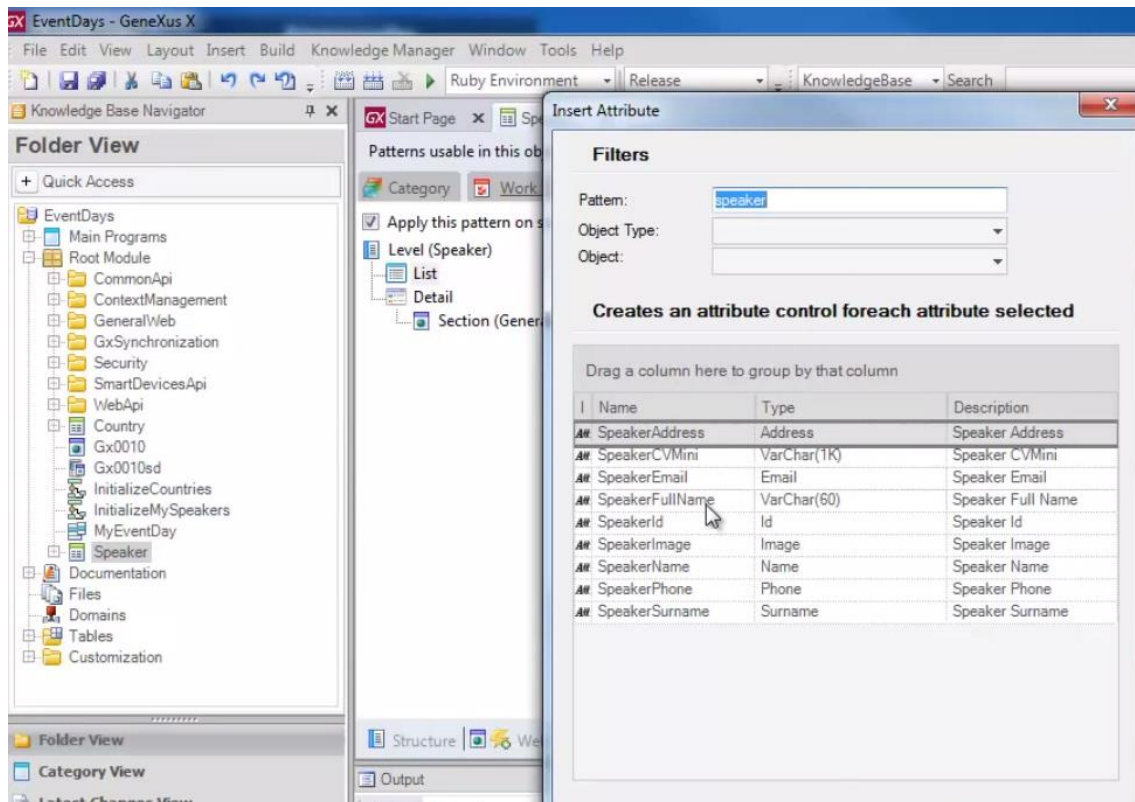


razón por la cual no está apareciendo una etiqueta para este control.

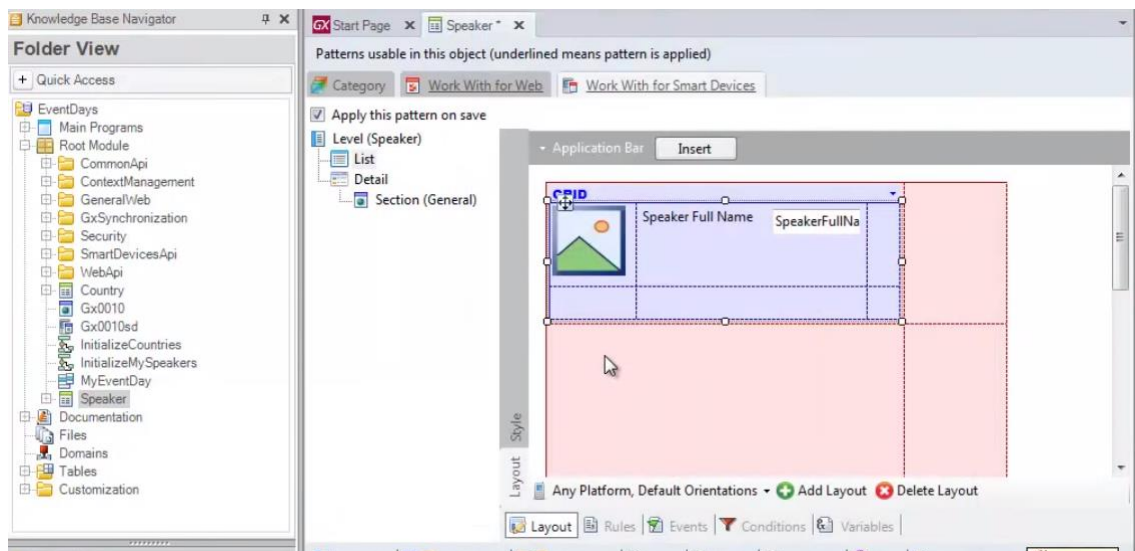
Vamos a sustituir este control correspondiente a **SpeakerName**, por el full name, el nombre completo del orador. Entonces borramos este control y con botón derecho sobre el interior del grid



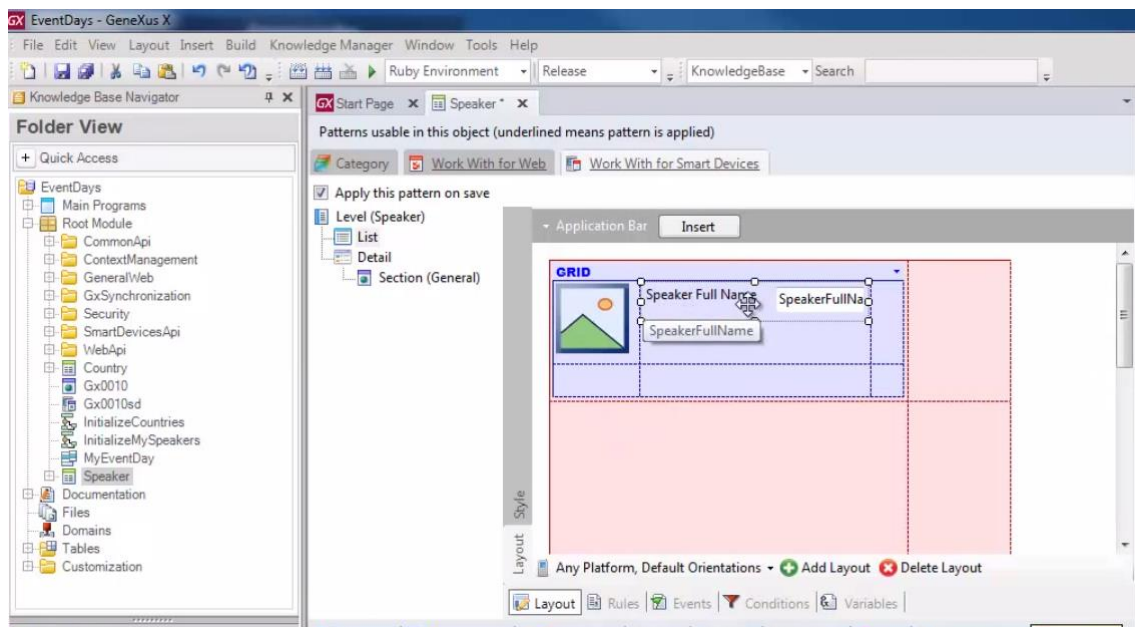
Hacemos Insert Attribute...



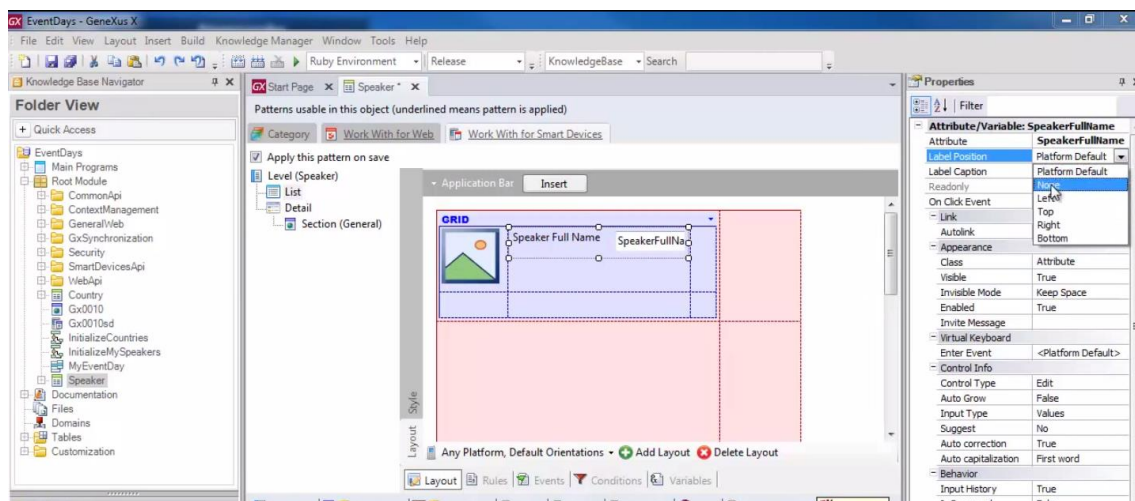
Y colocamos SpeakerFullName:



Observemos que ahora sí está apareciendo por defecto la etiqueta:



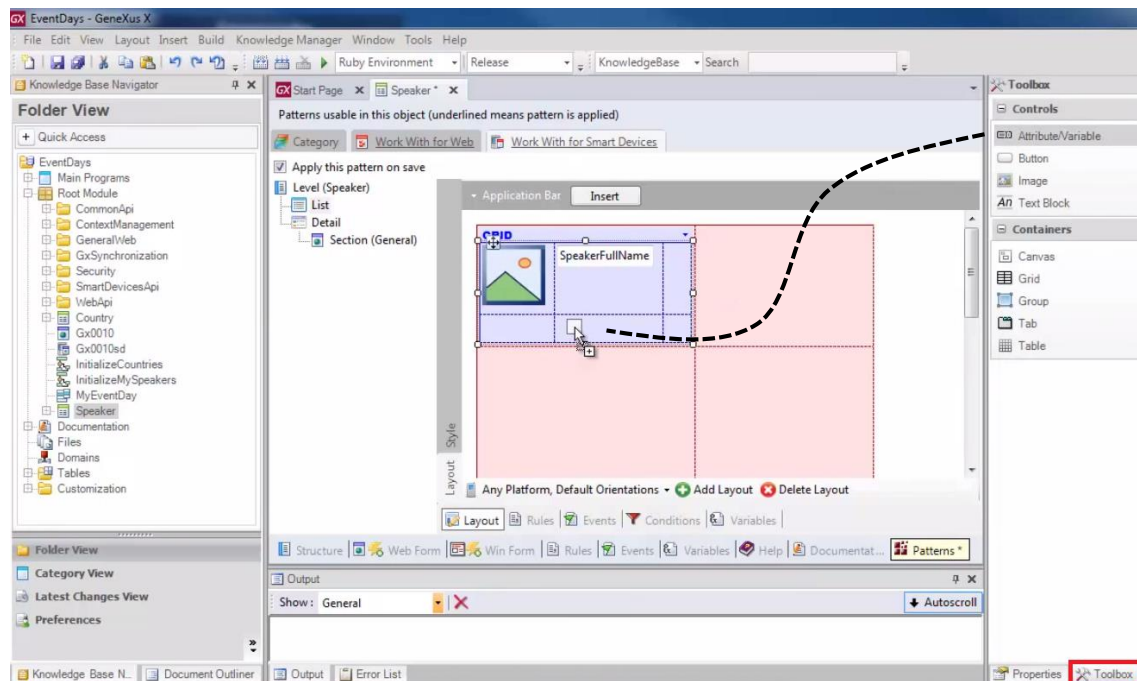
Si no la queremos, vamos a **Label Position** y modificamos el valor por defecto Platform Default por el valor: None



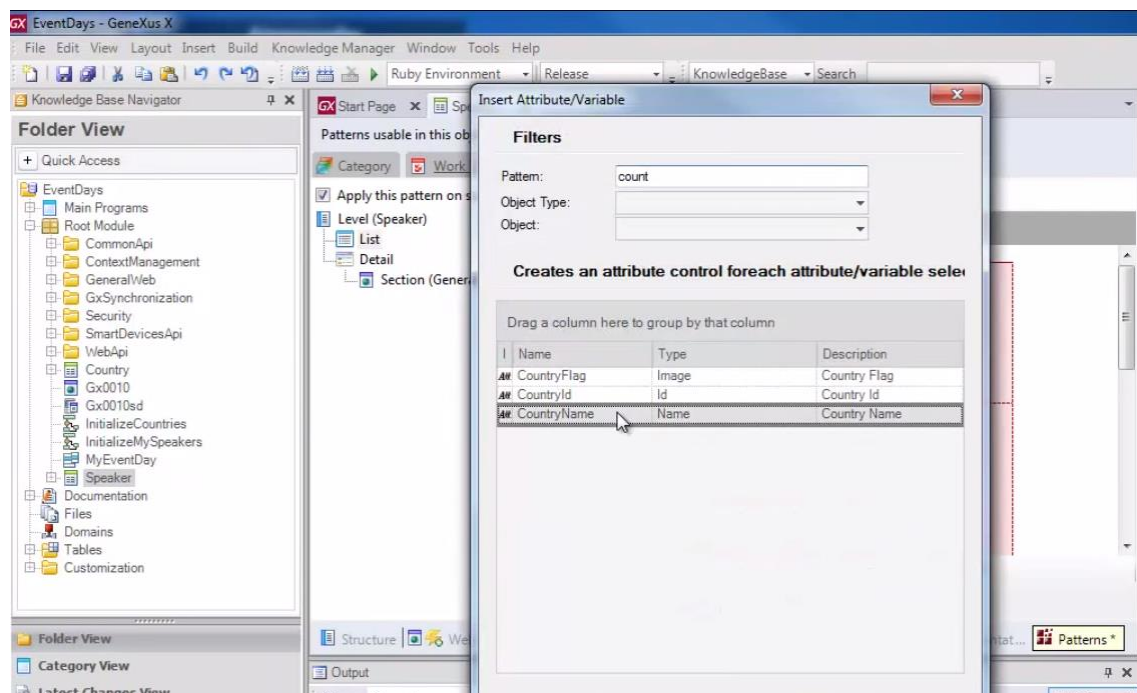
Video filmado con GeneXus X Evolution 3

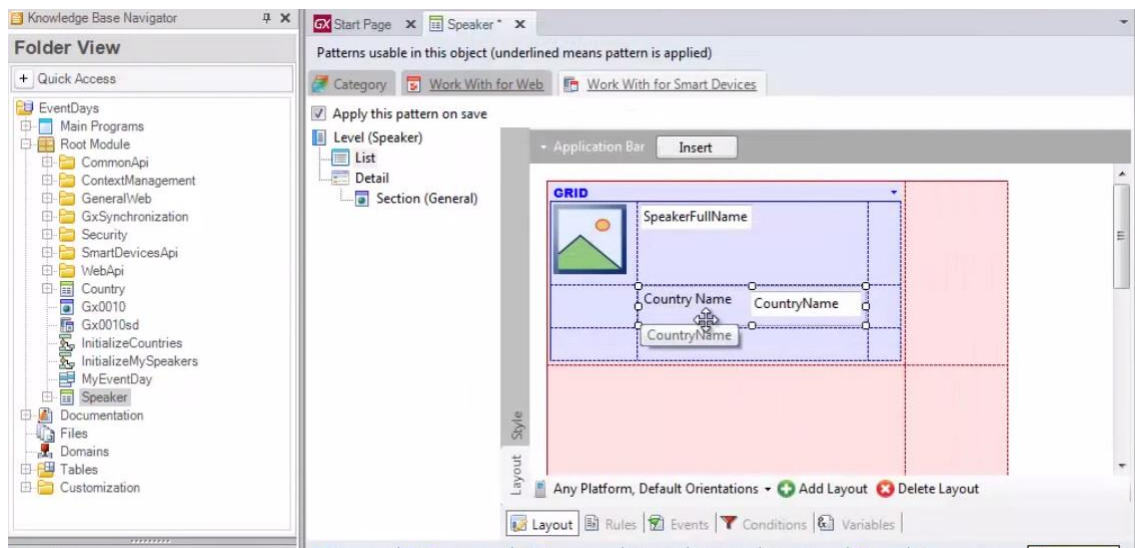
Ahora vamos a agregar el nombre del país del orador.

Otra vez botón derecho, o directamente en la Toolbox arrastramos el control atributo/variable

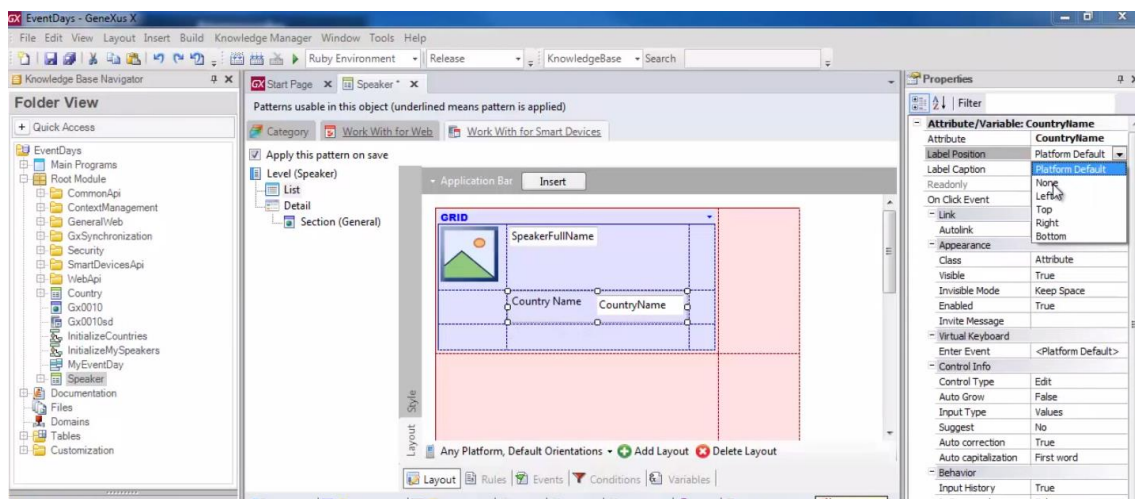


CountryName:

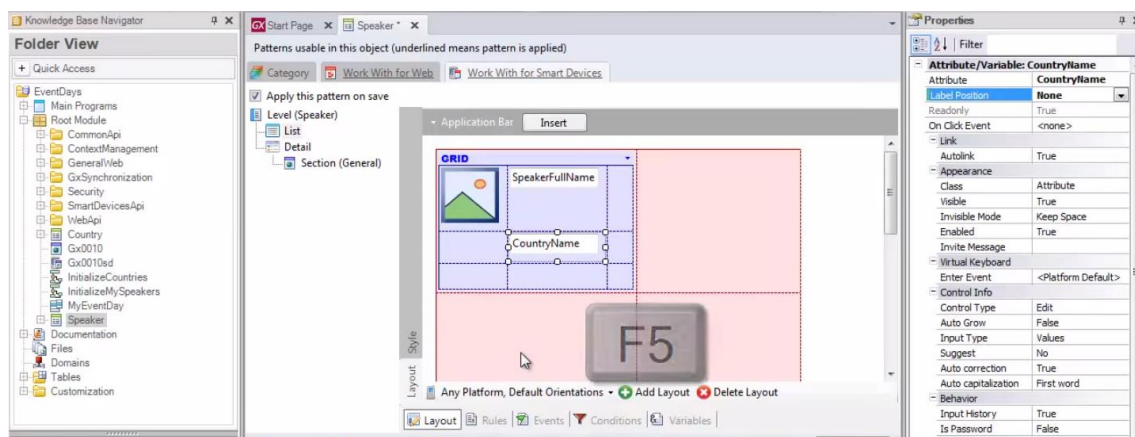


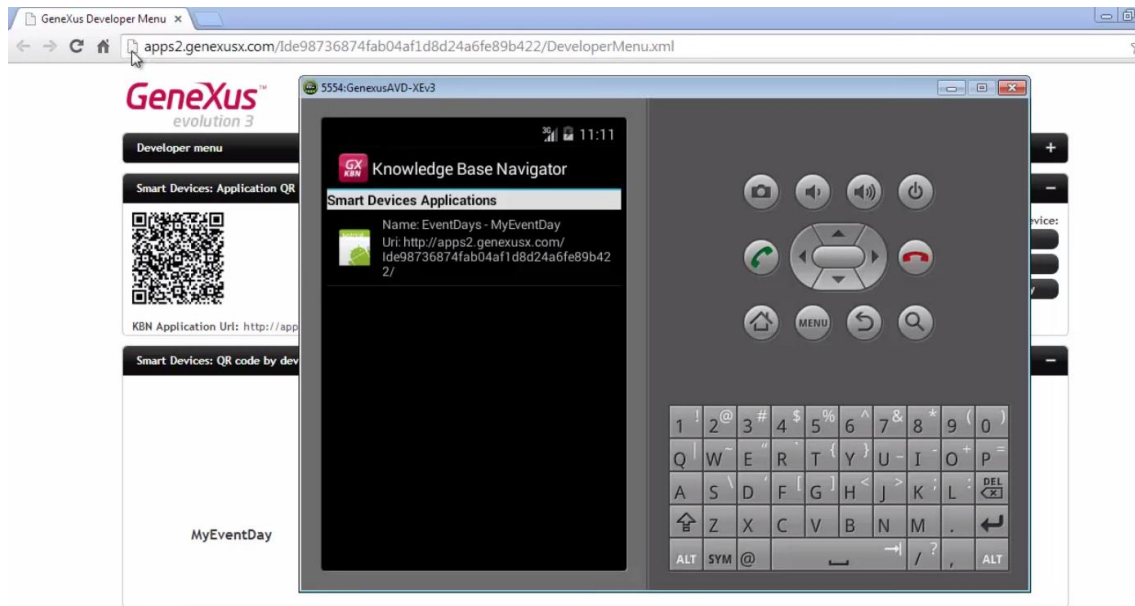


Y otra vez en las propiedades, a **Label Position** le cambiamos su valor por **None**

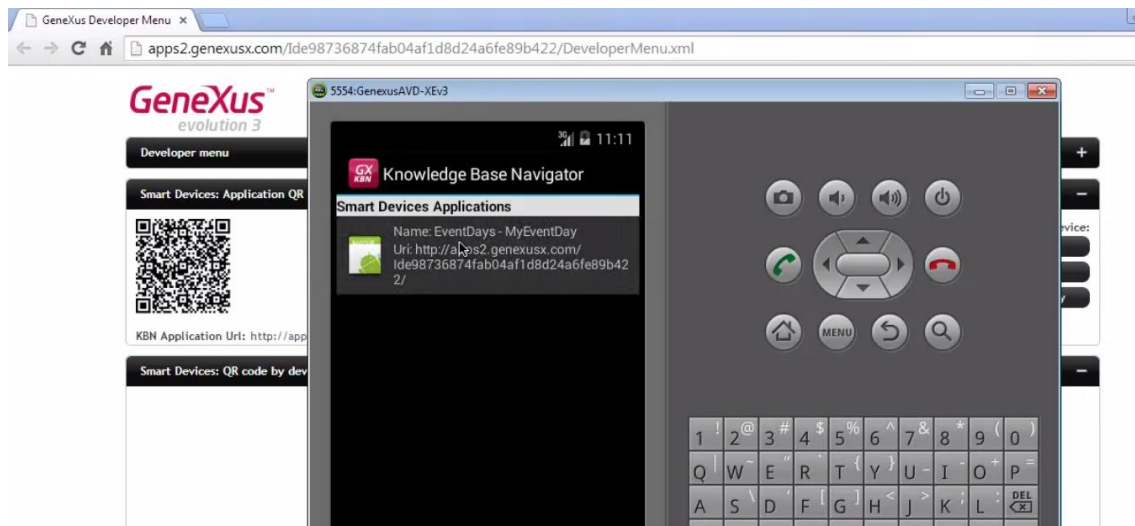


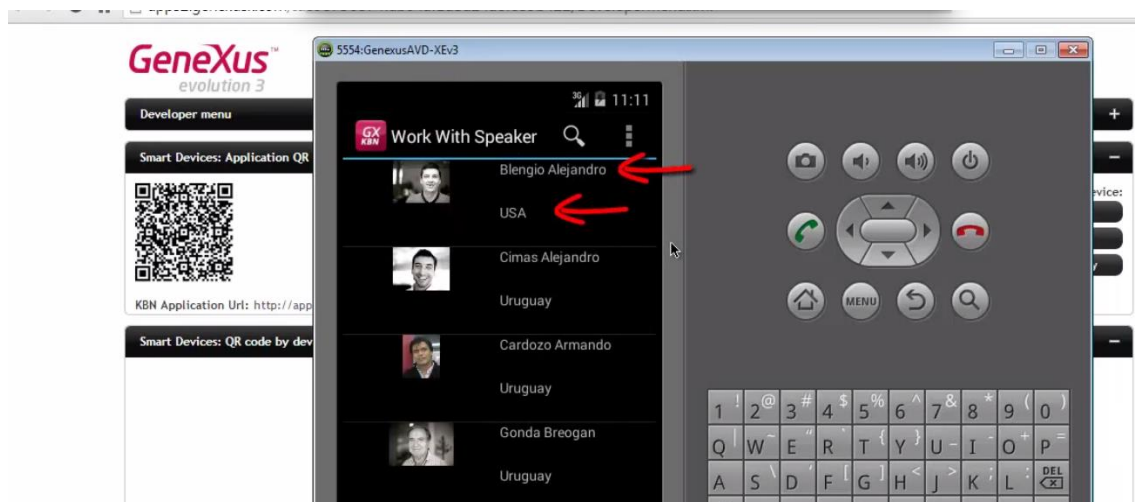
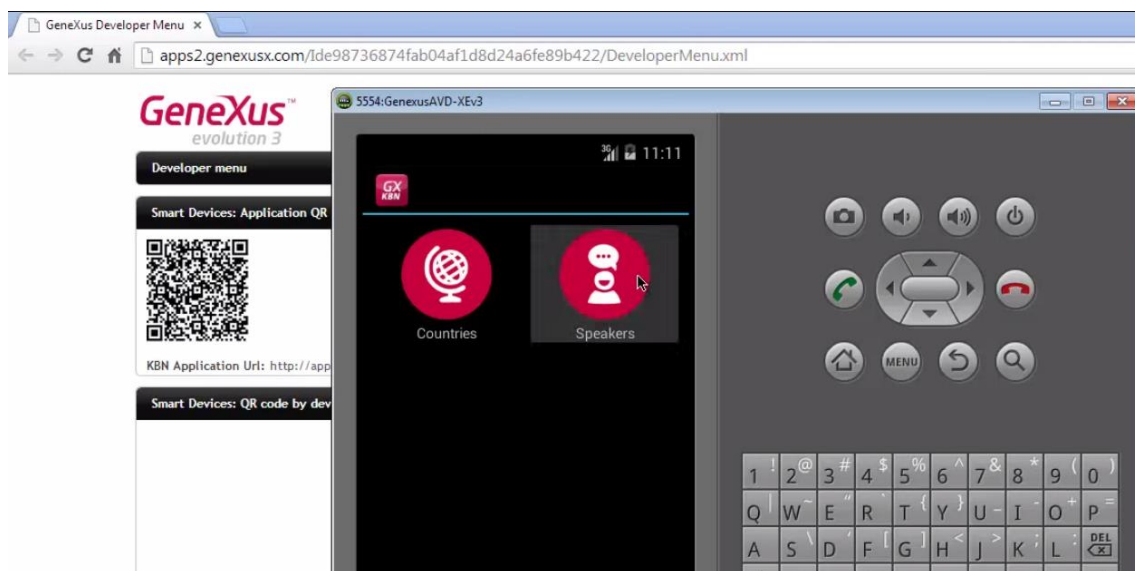
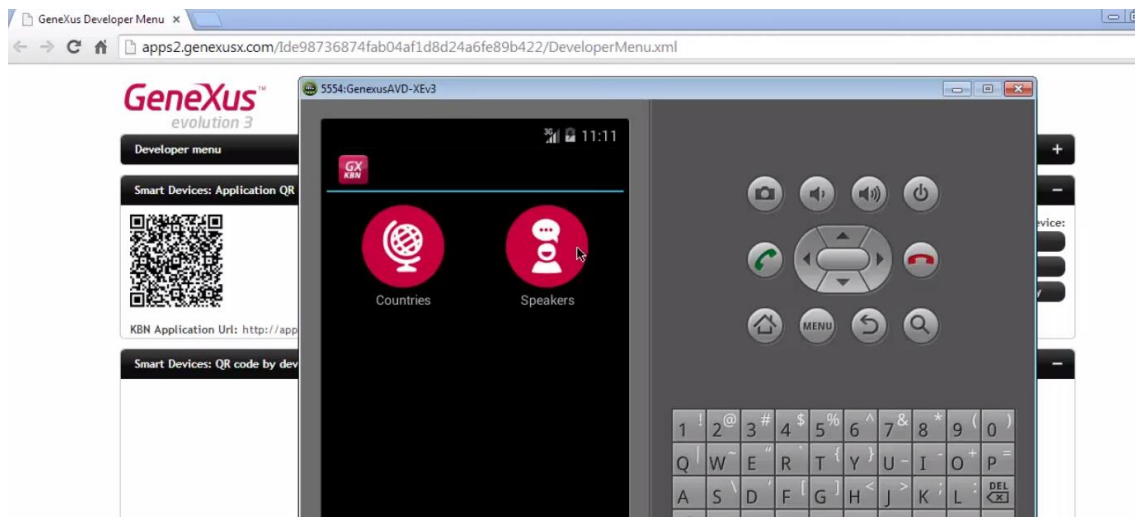
Hagamos un F5 para probar lo hecho:



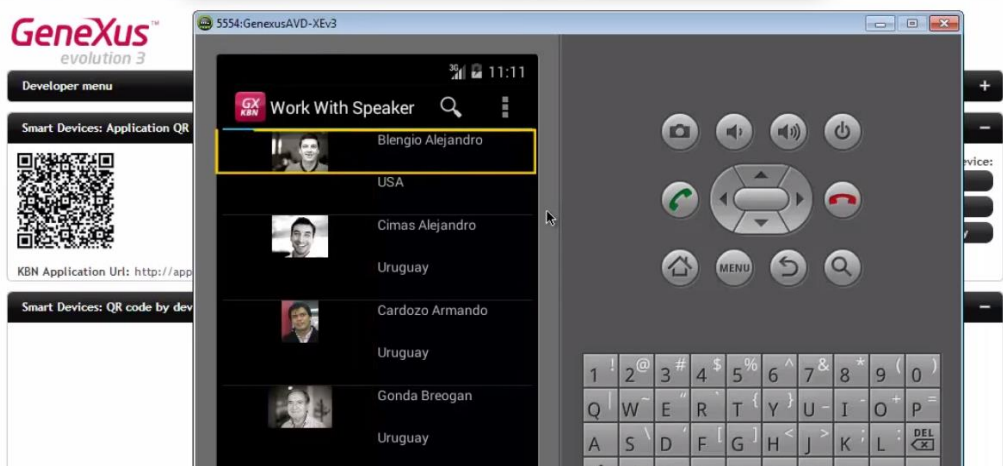
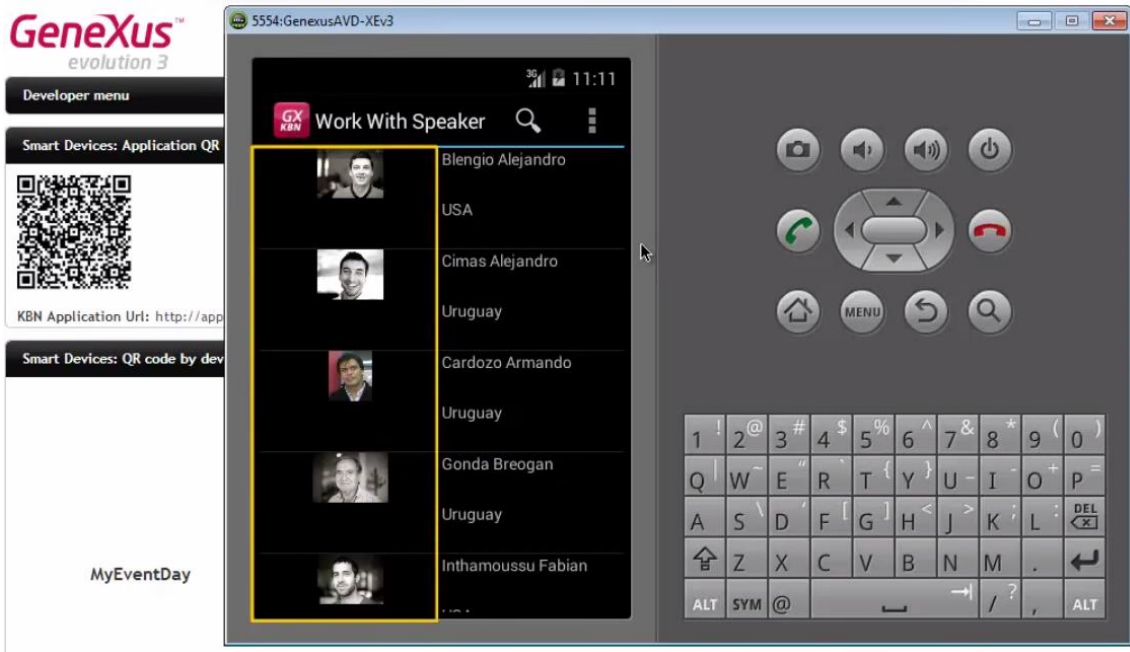


Como no tenemos Startup object definido, se abrió el Developer Menu web y el KBN.





Apreciamos que se puede dividir en 2 columnas y 2 filas por cada línea del grid:

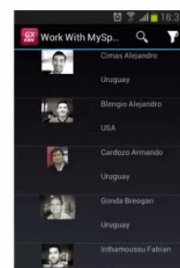
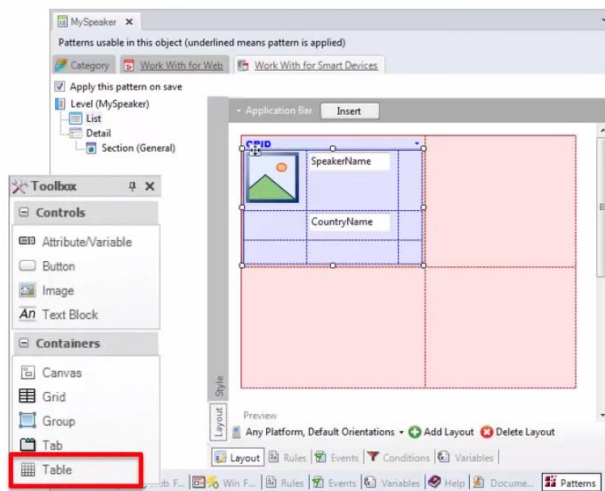




¿Cómo modificamos el lugar que ocupan los controles dentro de cada fila.. y sus distancias?
A través del trabajo con las tablas, que será central en las aplicaciones para Smart Devices.

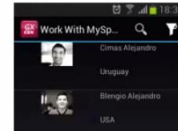
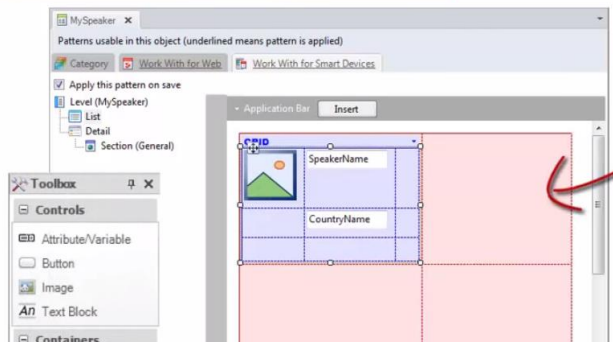


Controls in layouts: table



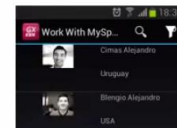
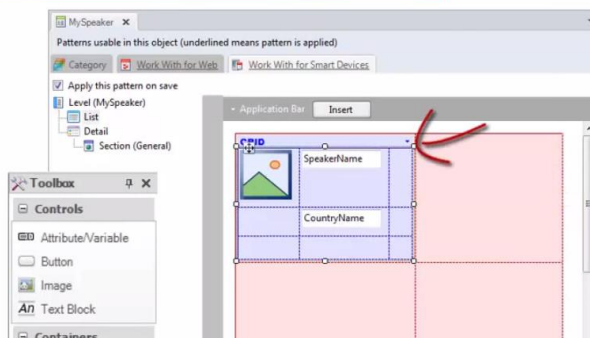
Todo Layout así esté vacío, tiene **una tabla raíz**:

Controls in layouts: table



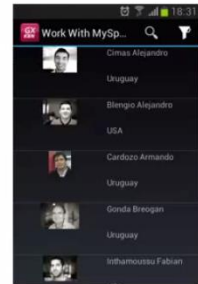
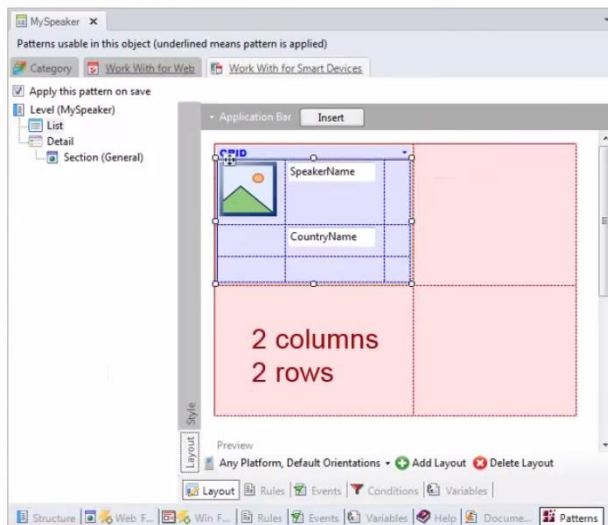
A su vez, todo grid va a definir una tabla para contener los controles de cada línea a ser cargada:

Controls in layouts: table



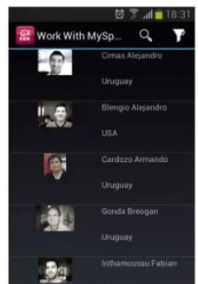
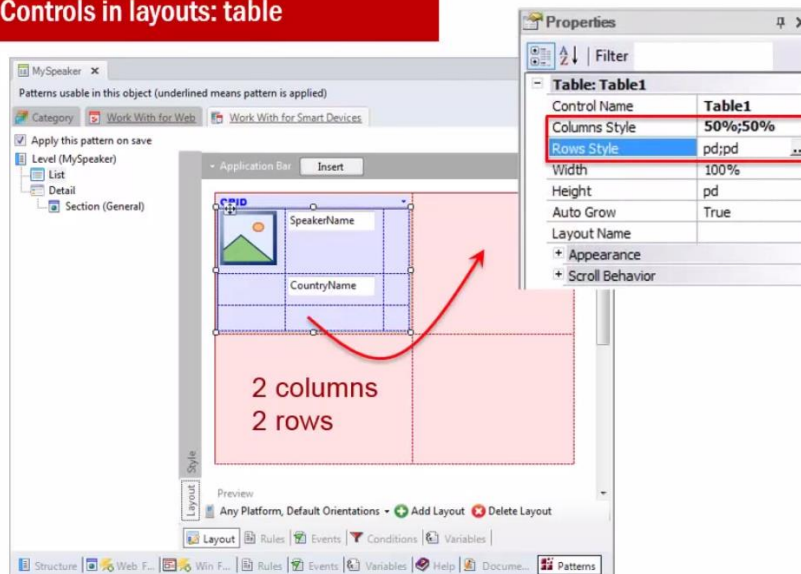
En este caso, la tabla tiene 2 columnas y 2 filas

Controls in layouts: table



Observemos las 2 propiedades: **Columns Style** y **Rows Style**:

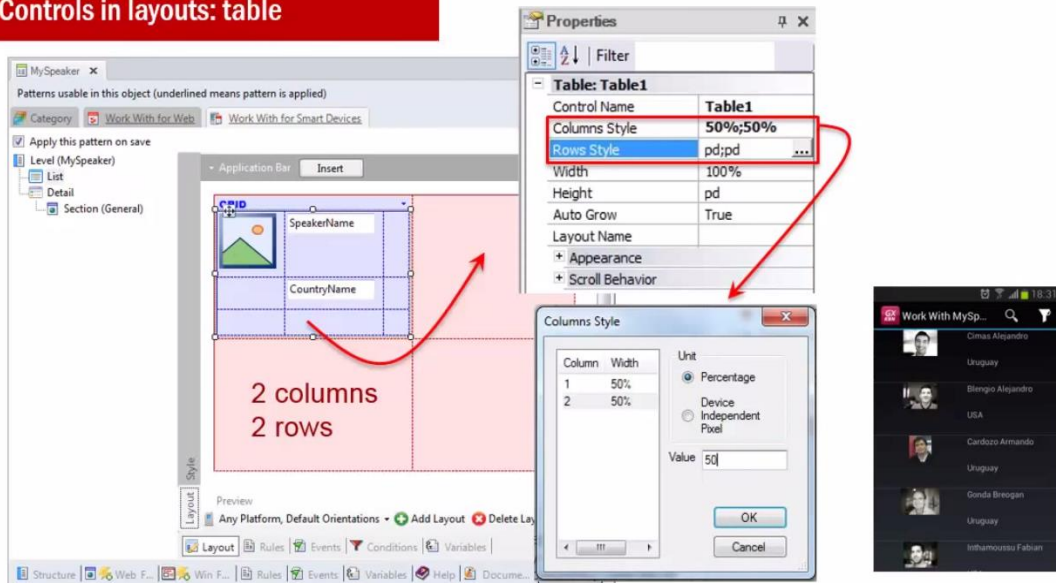
Controls in layouts: table



que permiten definir el tamaño que ocupará cada columna y cada fila de la tabla.

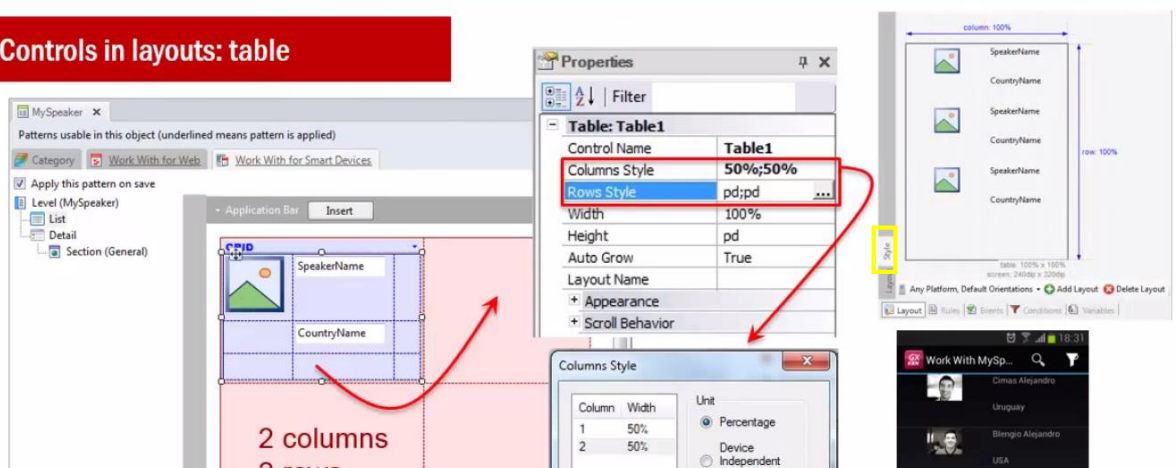
Observemos las columnas. Sus tamaños se pueden especificar en 2 unidades: Porcentaje o DIPs (Device Independent Pixel):

Controls in layouts: table



Observemos que tenemos la solapa “Style”, que nos muestra visualmente cómo se verá en ejecución

Controls in layouts: table



Si vemos las columnas, nos queda visualmente claro ese 50%:

Controls in layouts: table

MySpeaker x

Patterns usable in this object (underlined means pattern is applied)

Category: Work With for Web Work With for Smart Devices

Apply this pattern on save

Level (MySpeaker)

List

Detail

Section (General)

Application Bar

Insert

CPID

SpeakerName

CountryName

2 columns
2 rows

Preview

Any Platform, Default Orientations

Add Layout

Delete Layout

Layout

Rules

Events

Conditions

Variables

Structure

Web F...

Win F...

Rules

Events

Variables

Help

Docume...

Properties

Filter

Table: Table1

Control Name

Table1

Columns Style

50%;50%

Rows Style

pd;pd

Width

100%

Height

pd

Auto Grow

True

Layout Name

Appearance

Scroll Behavior

Columns Style

Column

Width

Unit

Percentage

Device Independent Pixel

Value

50

OK

Cancel

column: 100%

row: 100%

SpeakerName

CountryName

SpeakerName

CountryName

SpeakerName

CountryName

SpeakerName

CountryName

table: 100% x 100%

screen: 240dp x 320dp

Any Platform, Default Orientations

Add Layout

Delete Layout

Layout

Rules

Events

Conditions

Variables

Work With MySp...

Cinco Alejandro

Uruguay

Blanco Alejandro

USA

Cardozo Armando

Uruguay

Gonda Breogan

Uruguay

Inthamoussau Fabian

Controls in layouts: table

MySpeaker x

Patterns usable in this object (underlined means pattern is applied)

Category: Work With for Web Work With for Smart Devices

Apply this pattern on save

Level (MySpeaker)

List

Detail

Section (General)

Application Bar

Insert

CPID

SpeakerName

CountryName

2 columns
2 rows

Preview

Any Platform, Default Orientations

Add Layout

Delete Layout

Layout

Rules

Events

Conditions

Variables

Structure

Web F...

Win F...

Rules

Events

Variables

Help

Docume...

Properties

Filter

Table: Table1

Control Name

Table1

Columns Style

50%;50%

Rows Style

pd;pd

Width

100%

Height

pd

Auto Grow

True

Layout Name

Appearance

Scroll Behavior

Columns Style

Column

Width

Unit

Percentage

Device Independent Pixel

Value

50

OK

Cancel

column: 100%

row: 100%

SpeakerName

CountryName

SpeakerName

CountryName

SpeakerName

CountryName

SpeakerName

CountryName

table: 100% x 100%

screen: 240dp x 320dp

Any Platform, Default Orientations

Add Layout

Delete Layout

Layout

Rules

Events

Conditions

Variables

Work With MySp...

Cinco Alejandro

Uruguay

Blanco Alejandro

USA

Cardozo Armando

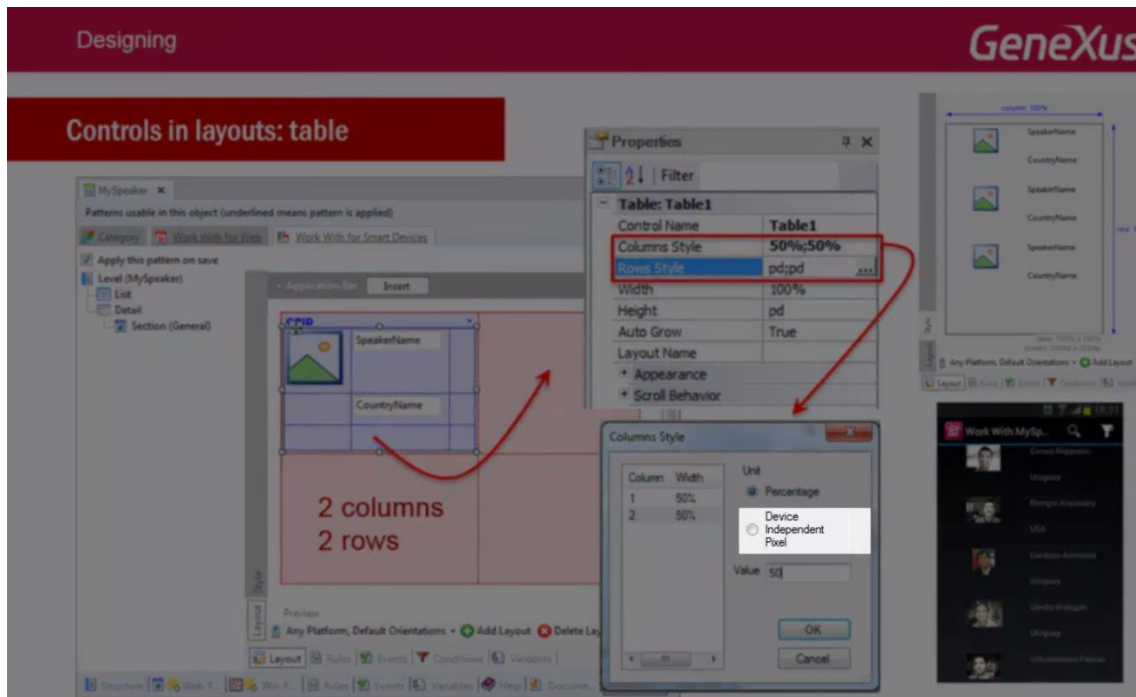
Uruguay

Gonda Breogan

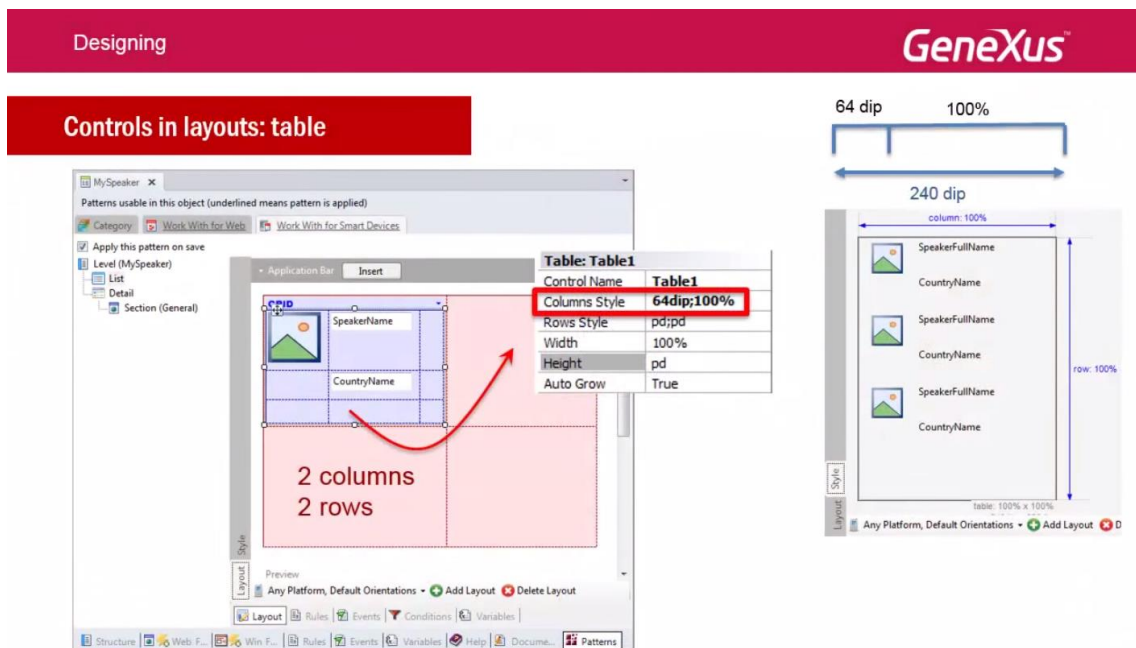
Uruguay

Inthamoussau Fabian

El **Device Independent Pixel** corresponde a la abstracción de 1 píxel, que luego una aplicación convierte a píxeles físicos, lo que permite escalar a diferentes tamaños de pantalla:



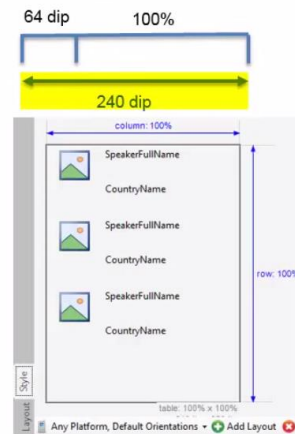
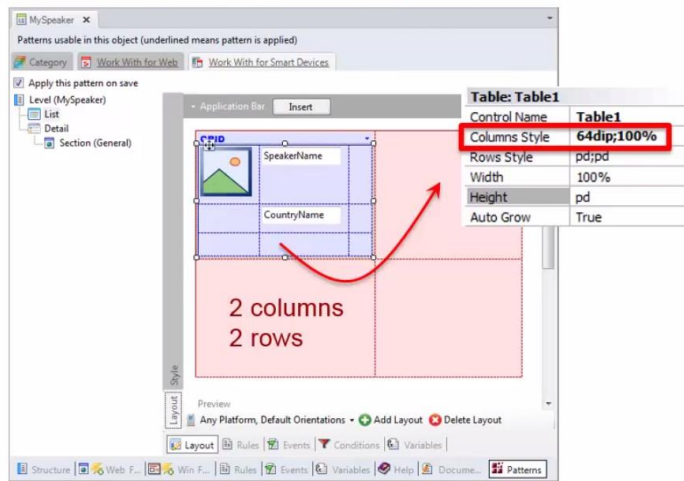
El DIP para cada plataforma tiene diferente número de píxeles.



Los porcentajes son relativos al valor que resulta de restar del ancho total, los valores fijos en DIPs.

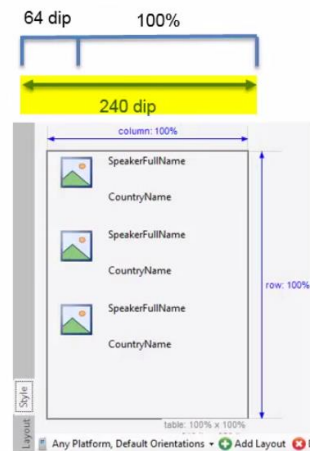
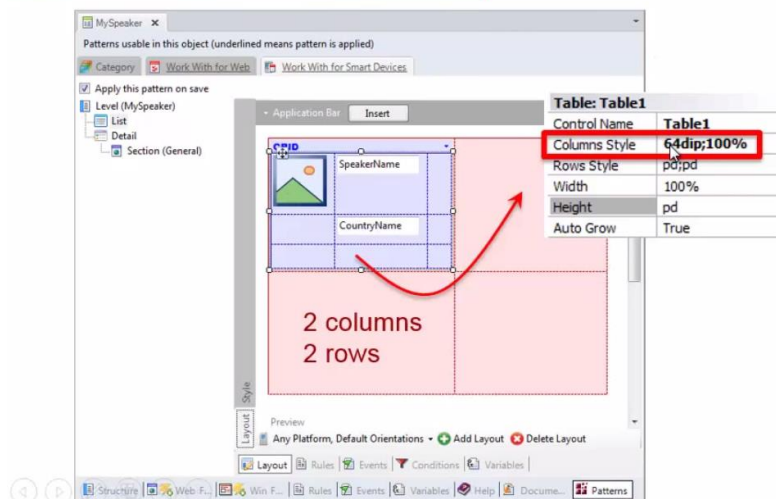
Así, si el ancho total es de 240 DIPs

Controls in layouts: table



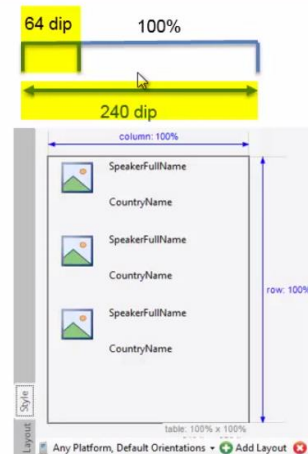
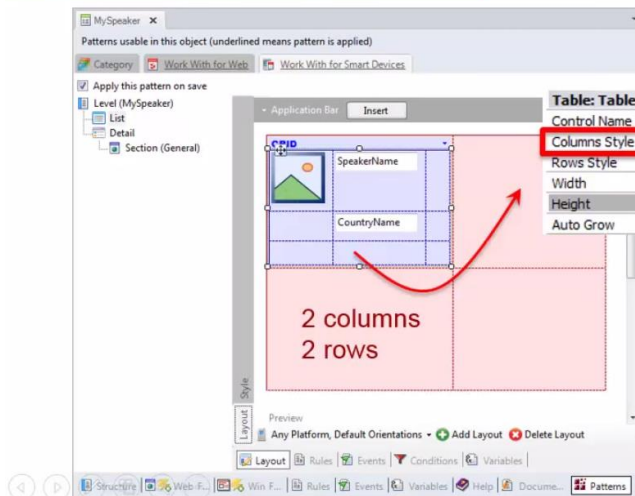
Y hay una columna de 64 DIPs

Controls in layouts: table



El valor de 100% para la siguiente, significa que esa segunda columna, ocupará el 100% restante

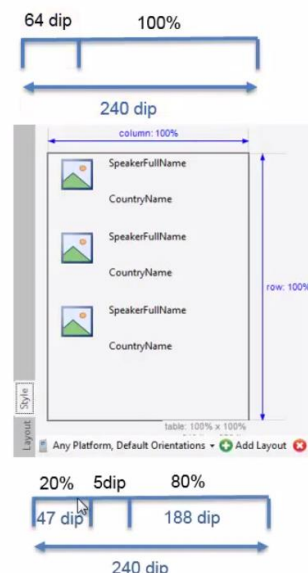
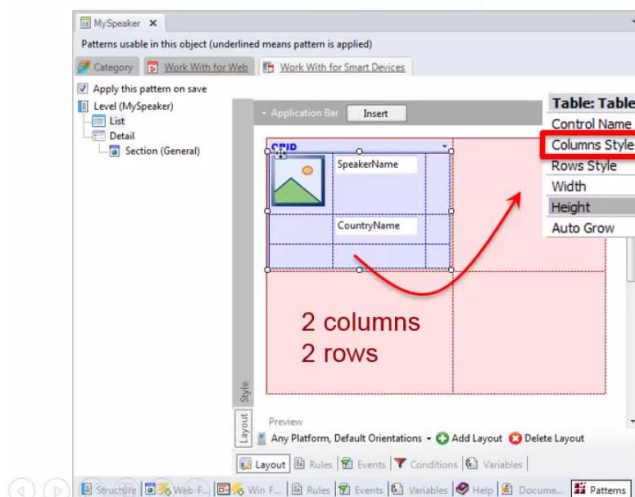
Controls in layouts: table



Es decir: 240 DIPS - 64 DIPS → 176 DIPS.

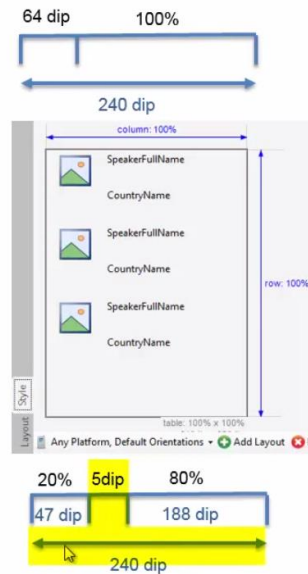
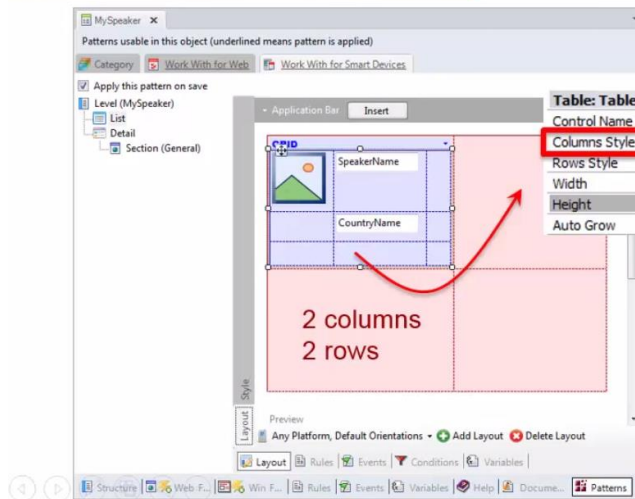
Si tuviéramos 3 columnas:

Controls in layouts: table



La primera con 20%, la segunda con valor fijo 5 DIPS, y la tercera con 80%, los valores que asumirán la primera y la tercera, se obtienen de aplicar esos porcentajes, al valor resultante de sustraer la suma de los valores fijos. Aquí sólo uno: 5 DIPS, del ancho de la tabla:

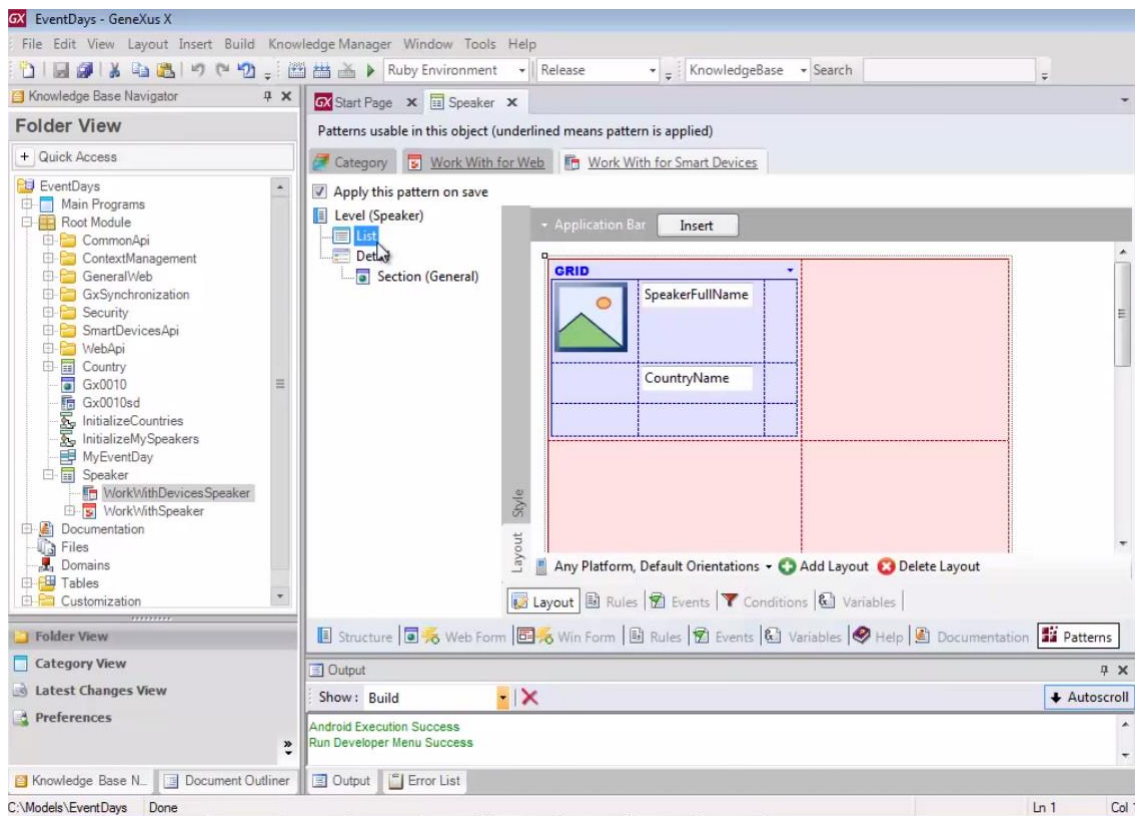
Controls in layouts: table



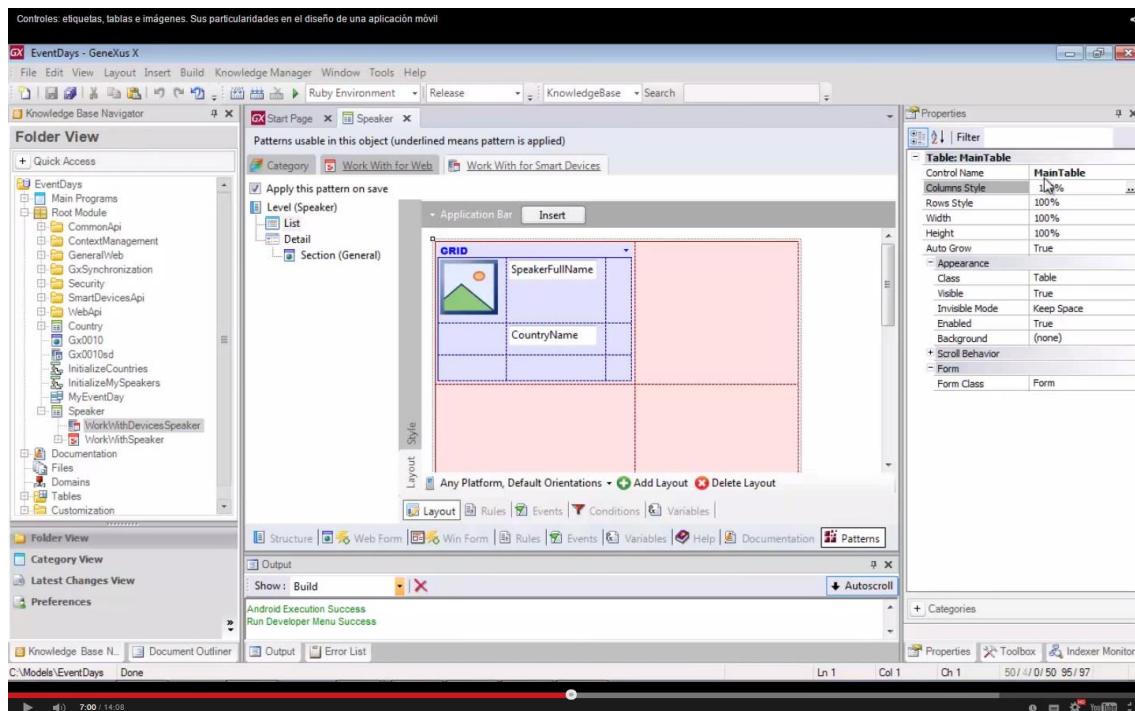
La segunda columna con 5 DIPs, puede usarse como espacio en blanco entre la primera y la tercera.

Personalicemos esto en GeneXus.

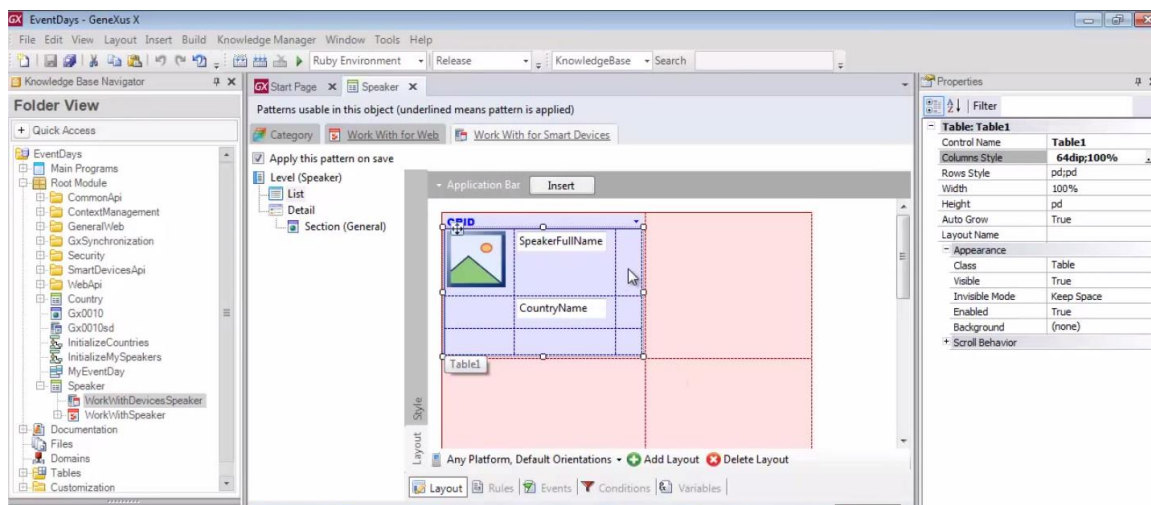
Vemos que para el List entonces tenemos



La tabla principal: **Main Table**

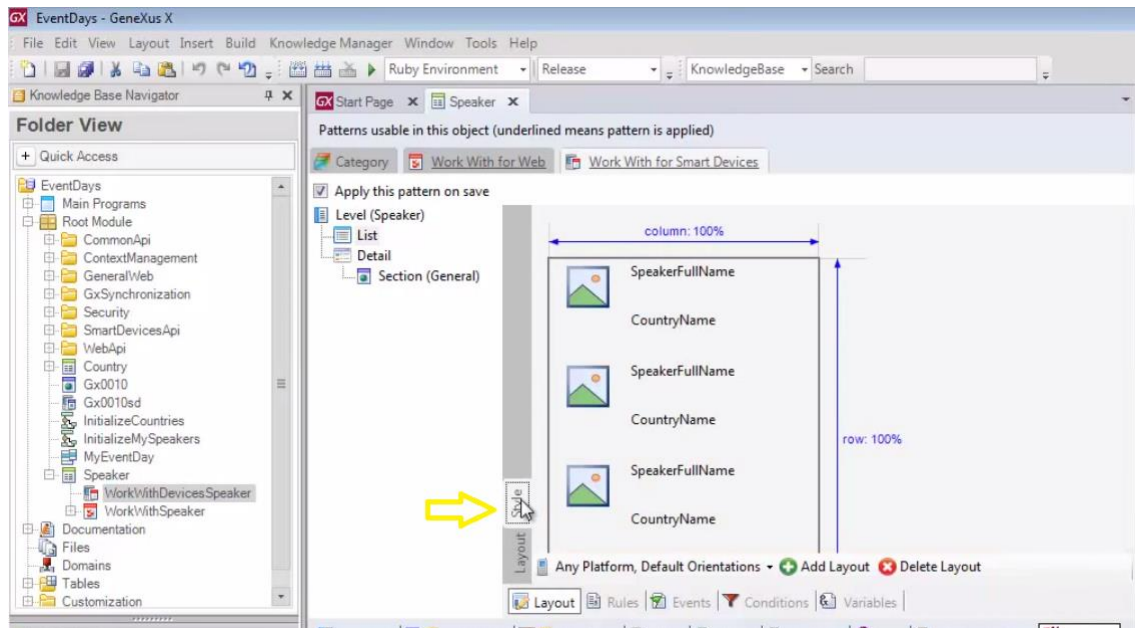


Y la tabla correspondiente al grid: **Table1**



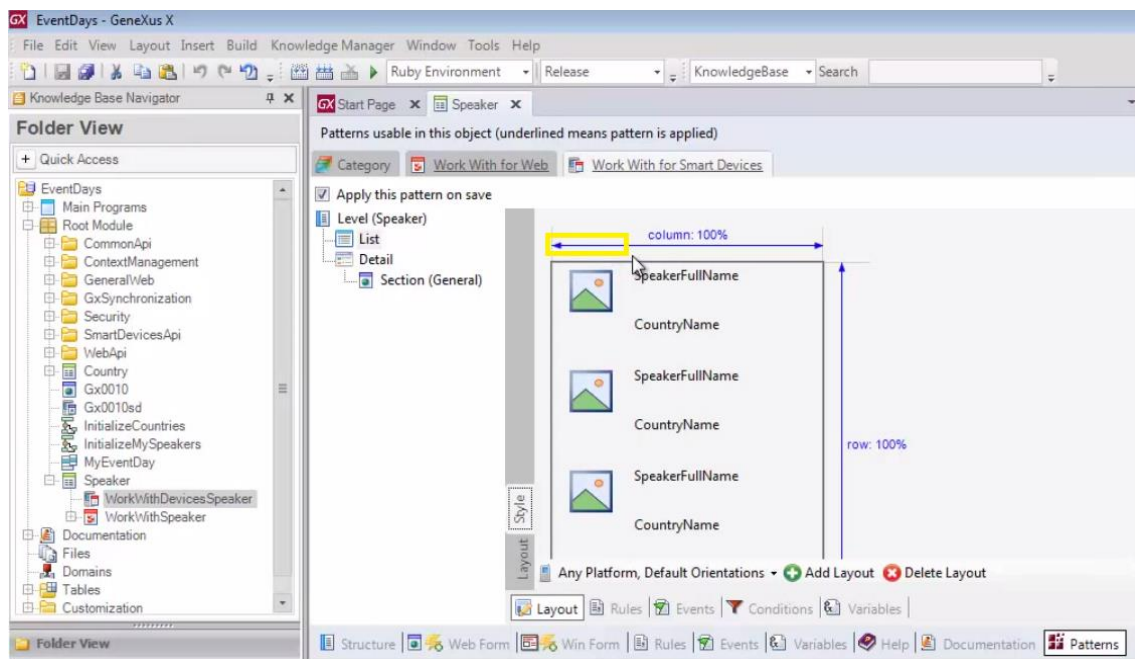
Si observamos las columnas, en este caso tenemos que la primera ocupa 64 DIPs (la de la imagen del orador) y la segunda el 100% restante.

Para ver gráficamente cómo lucirá el form en ejecución, vamos a la solapa “Style”:

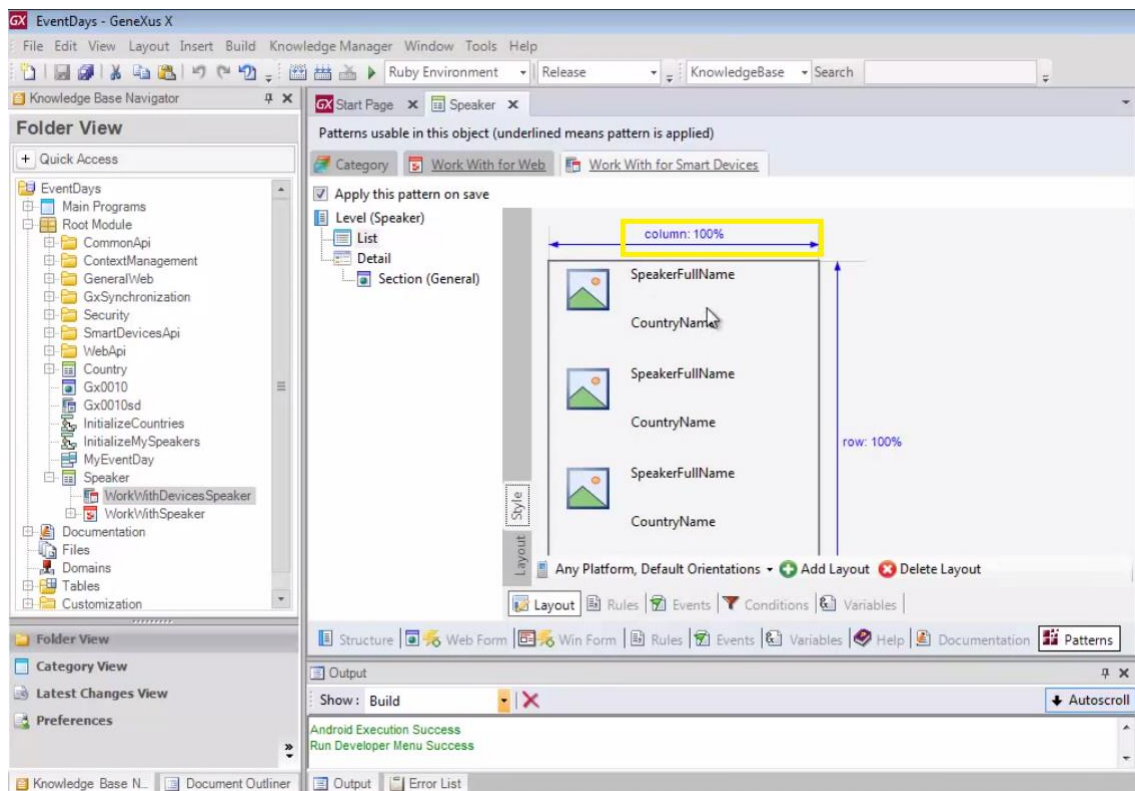


Y aquí vemos básicamente esa división en columnas, con sus tamaños.

64 DIPs:

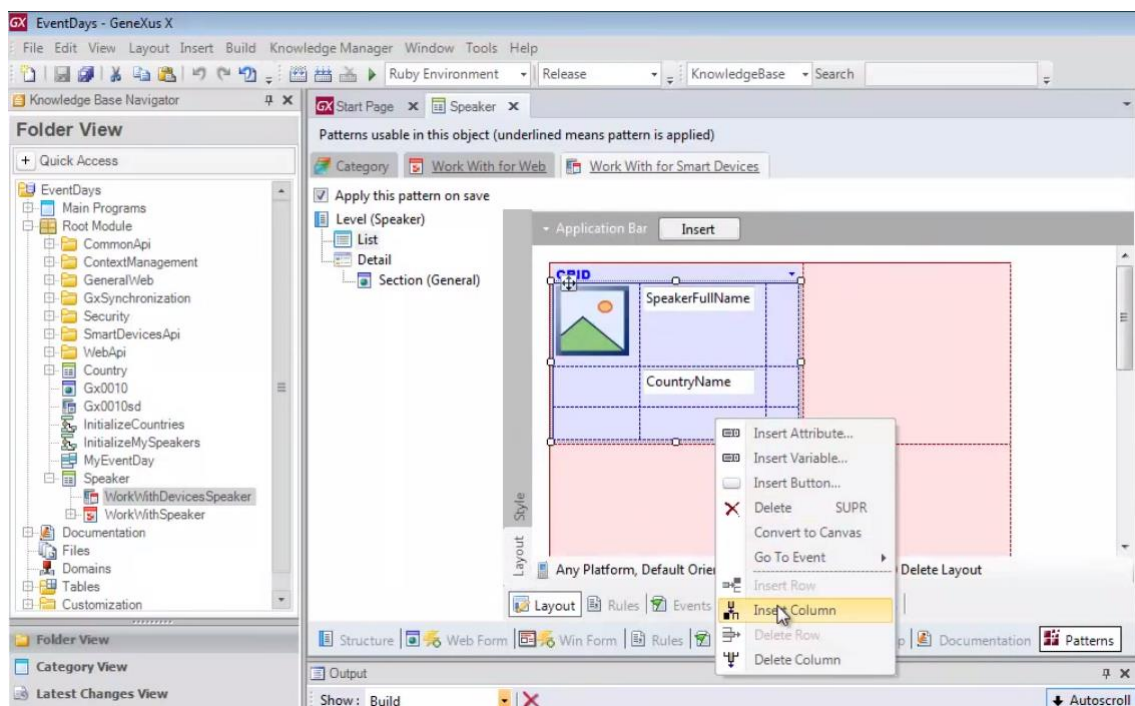


Y el 100% restante:

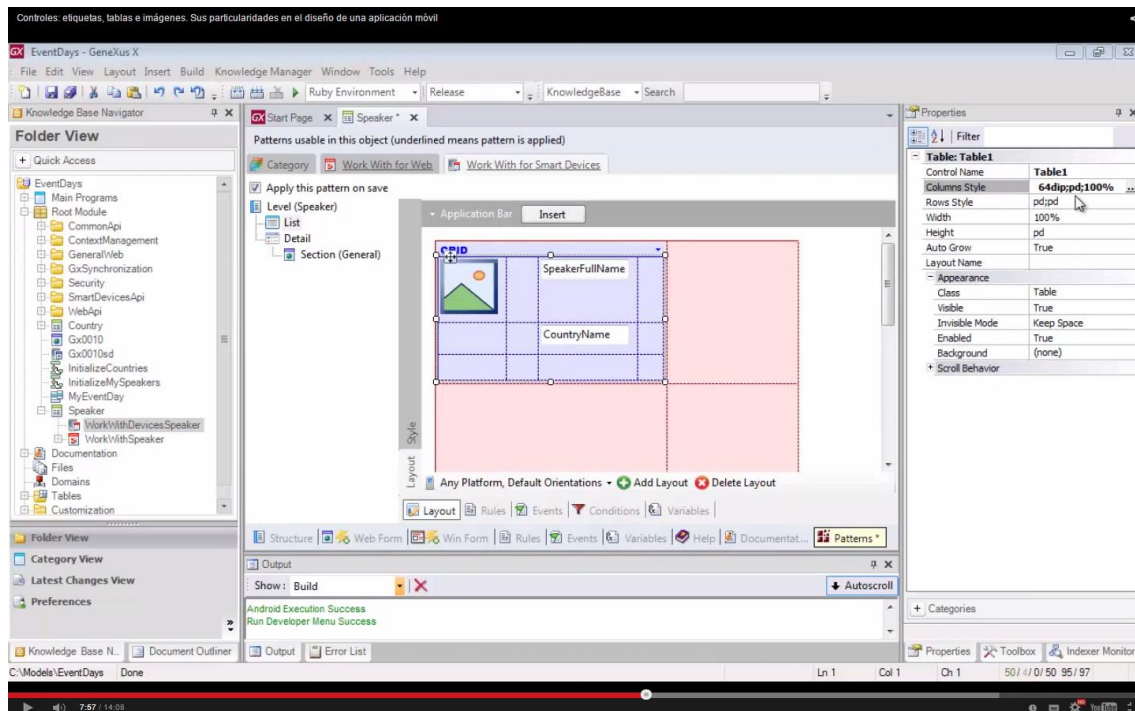


Vamos a agregar una columna entre medio de la primera y la segunda, para tener cierto “aire” entre los contenidos de cada una.

Así, botón derecho, **Insert Column**

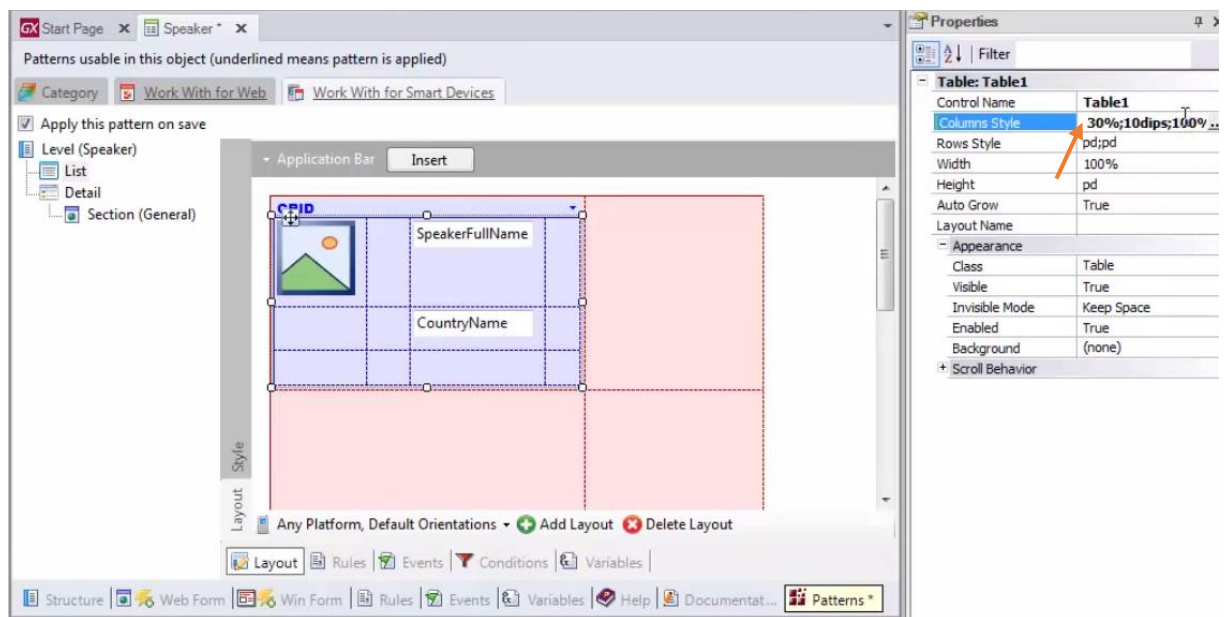


Y si vemos los anchos:

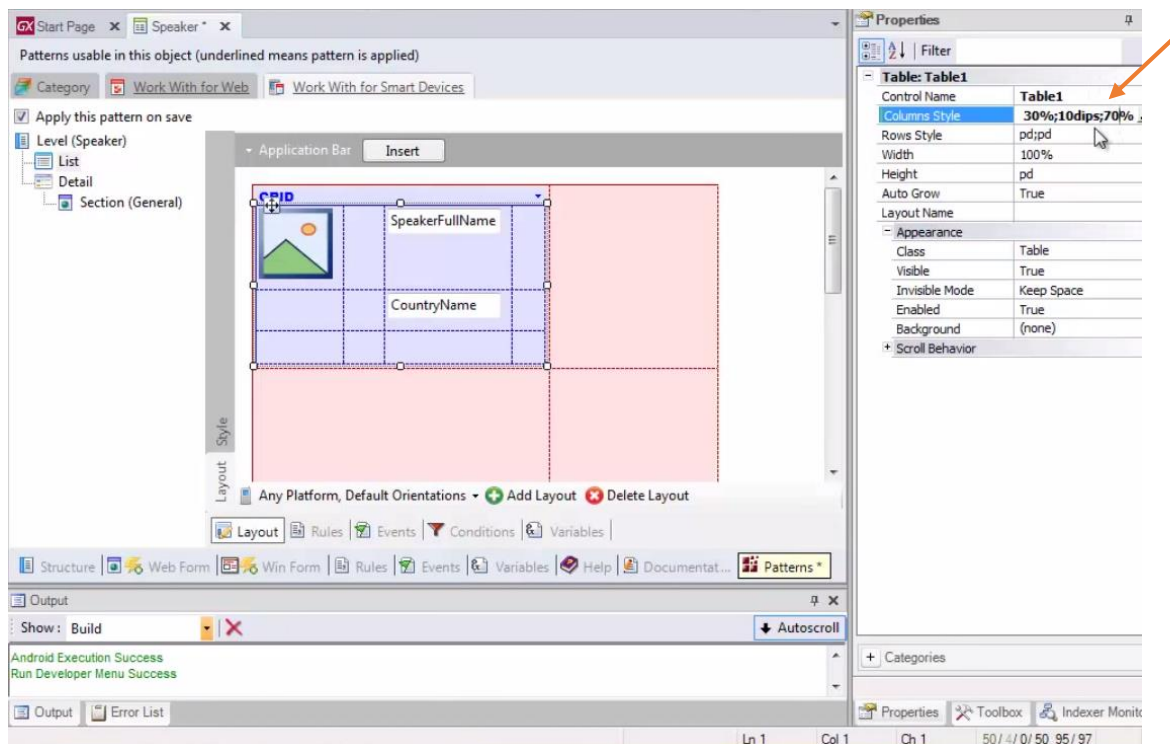


A esa columna del medio le asignó un tamaño: pd (que veremos luego).

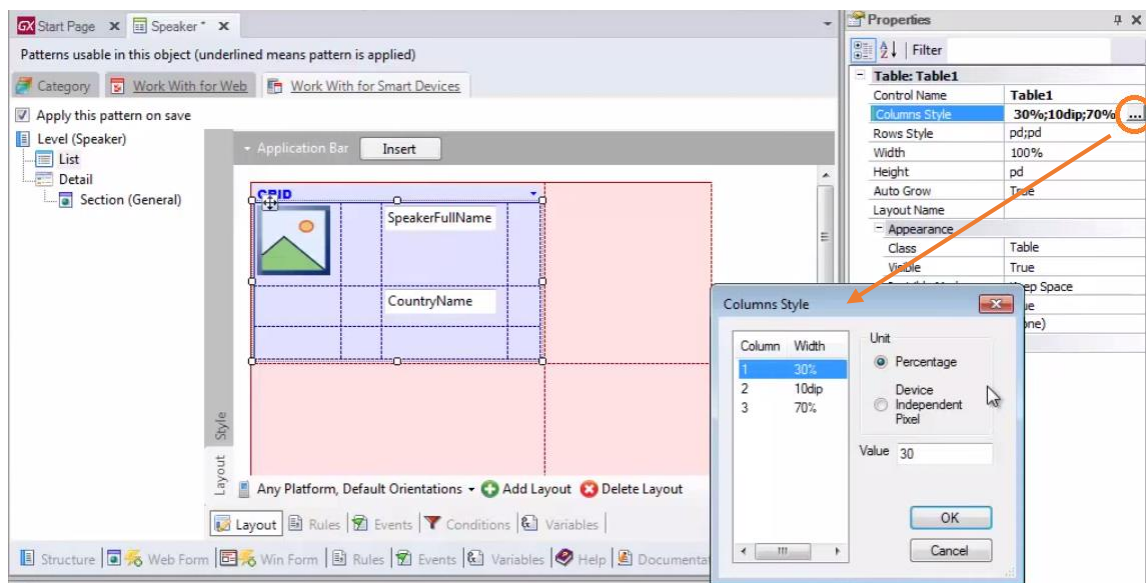
Vamos a modificarlo porque queremos que esa columna ocupe un tamaño de 10 DIPs (tamaño fijo) y que la primera columna, ocupe el 30% del ancho restante



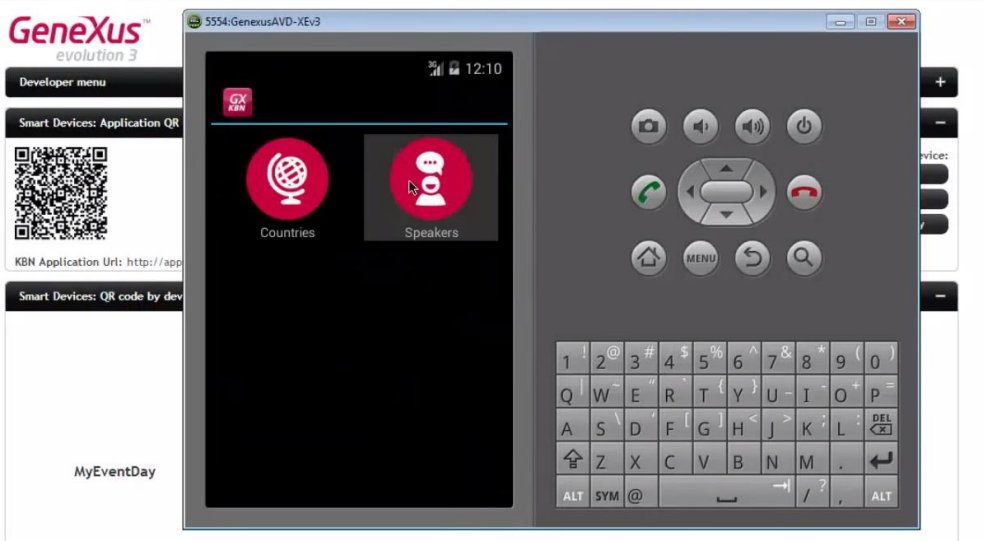
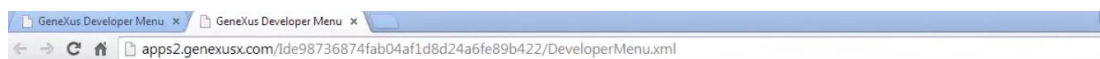
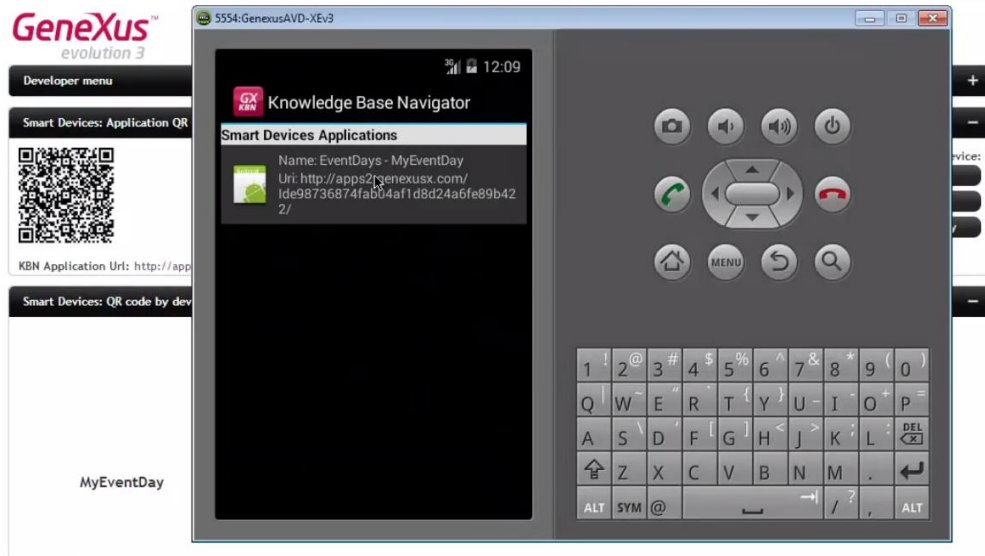
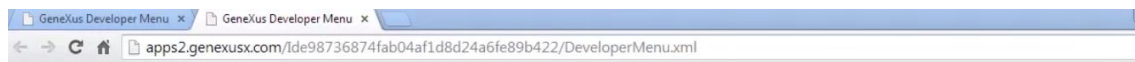
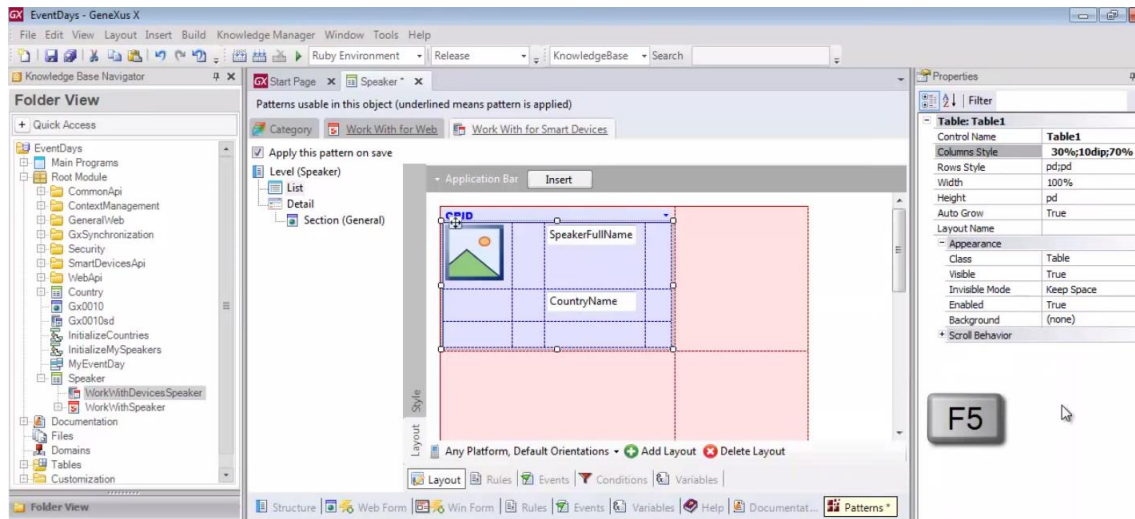
Y la tercera, lógicamente el 70%.

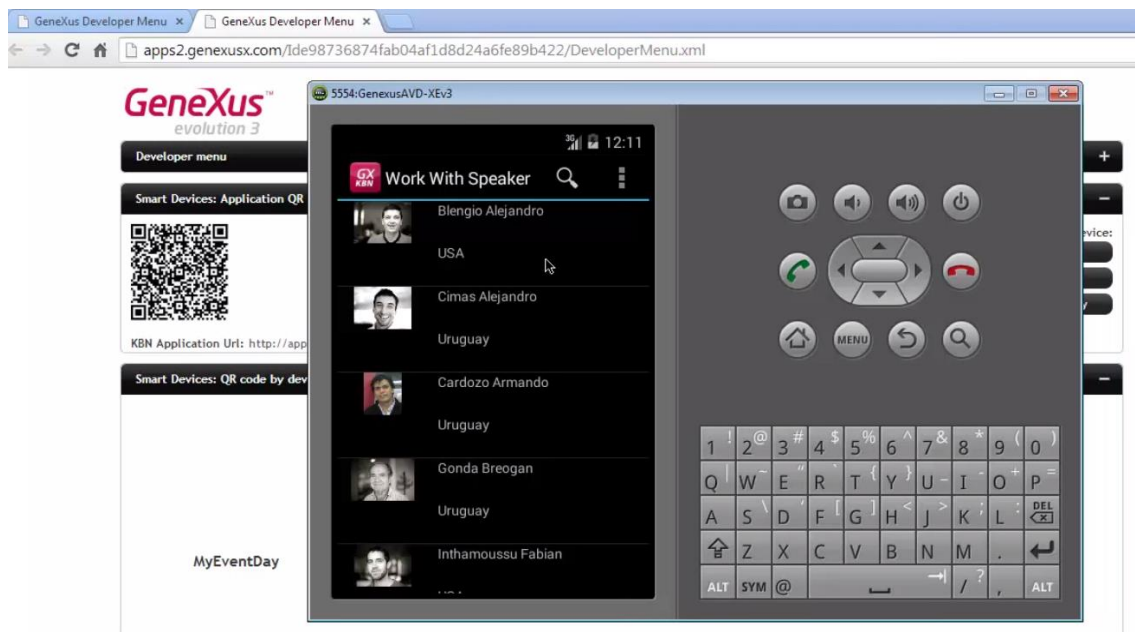


Podemos modificar directamente escribiendo en la propiedad, o abriendo esta ventana:

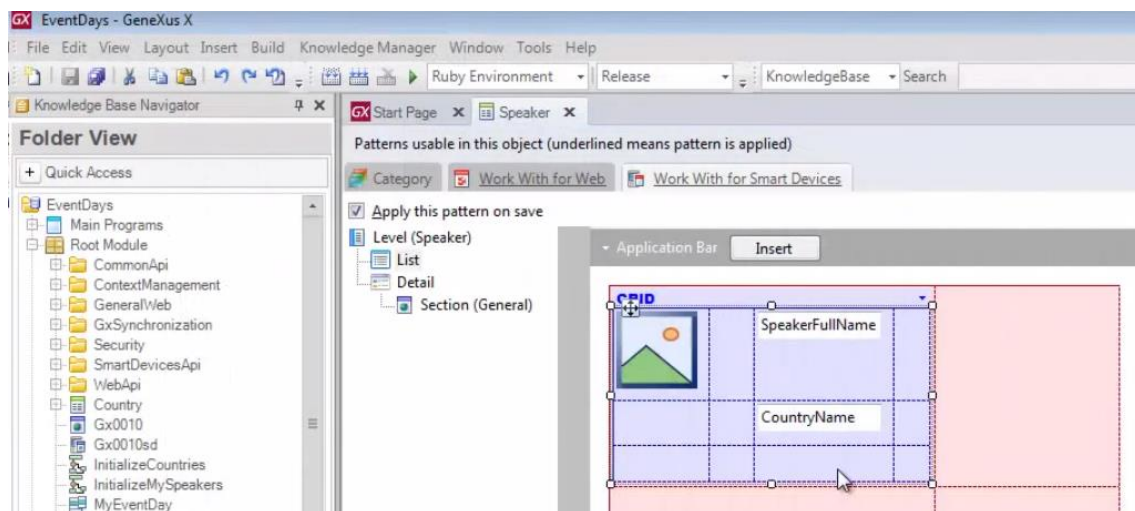


Ejecutemos: F5..

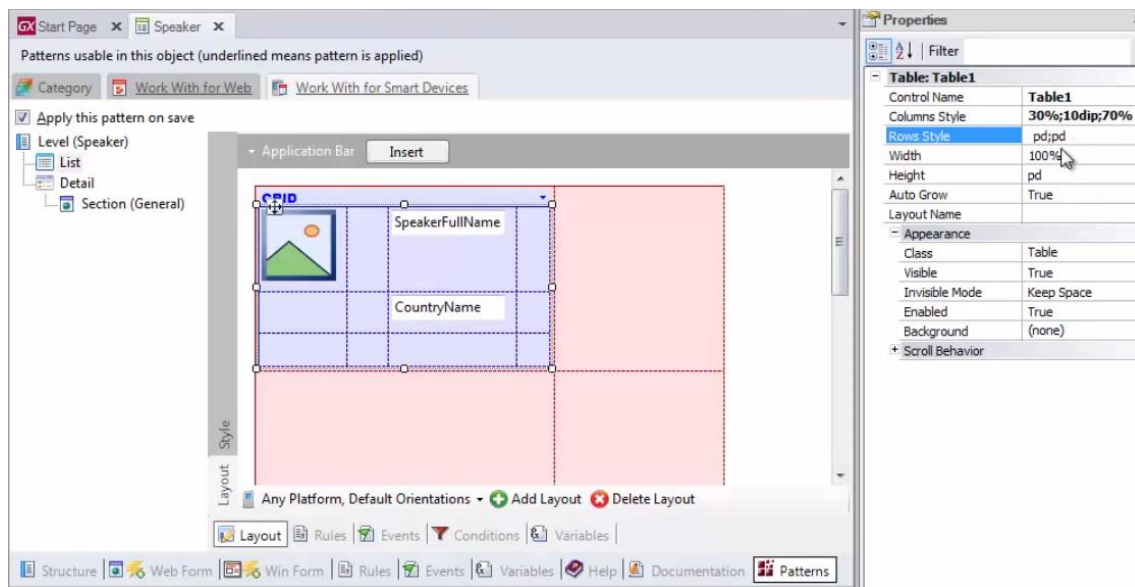




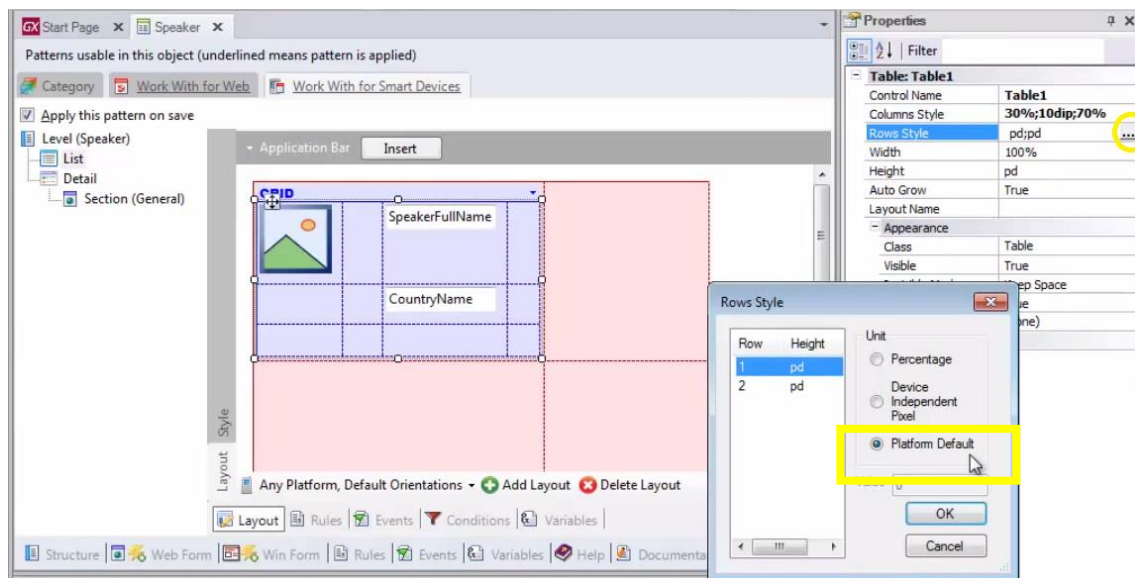
Si ahora observamos las filas:



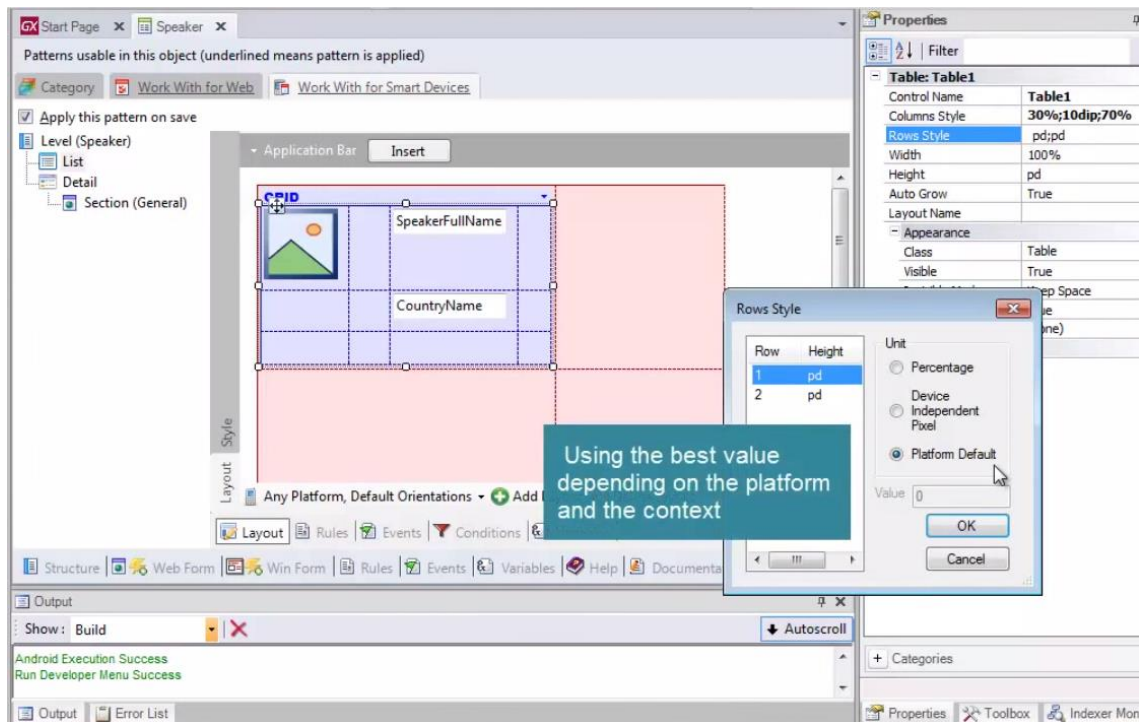
vemos que tenemos 2 con el valor: pd



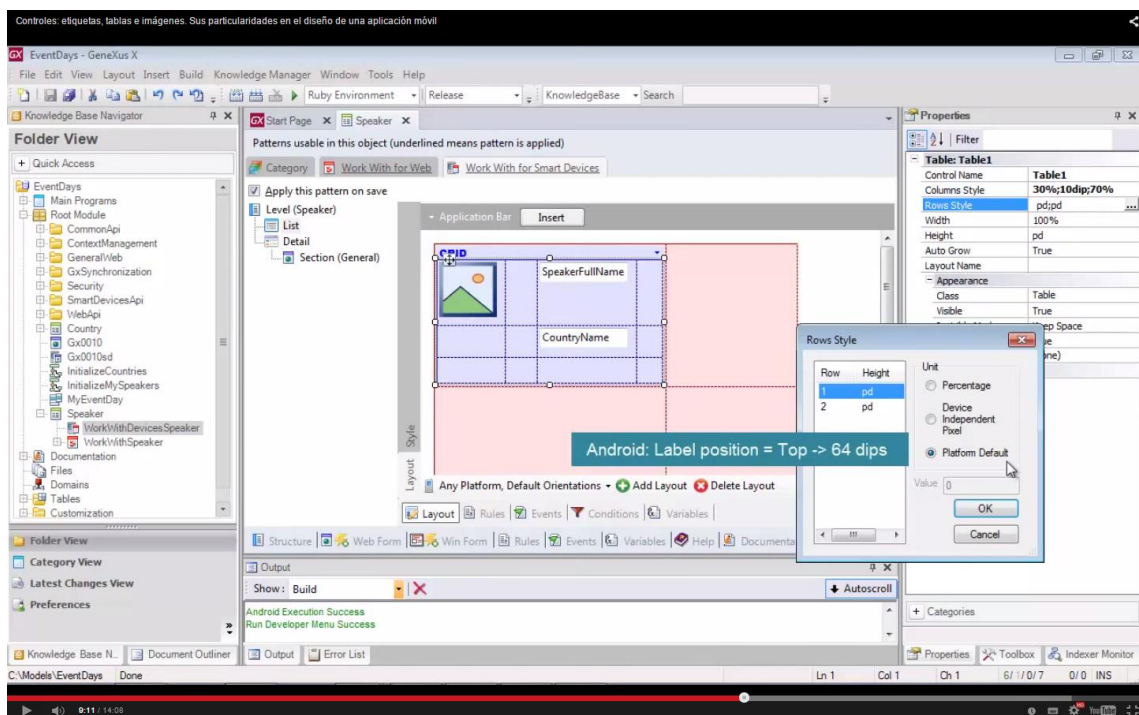
Que corresponde a la unidad **Platform Default**



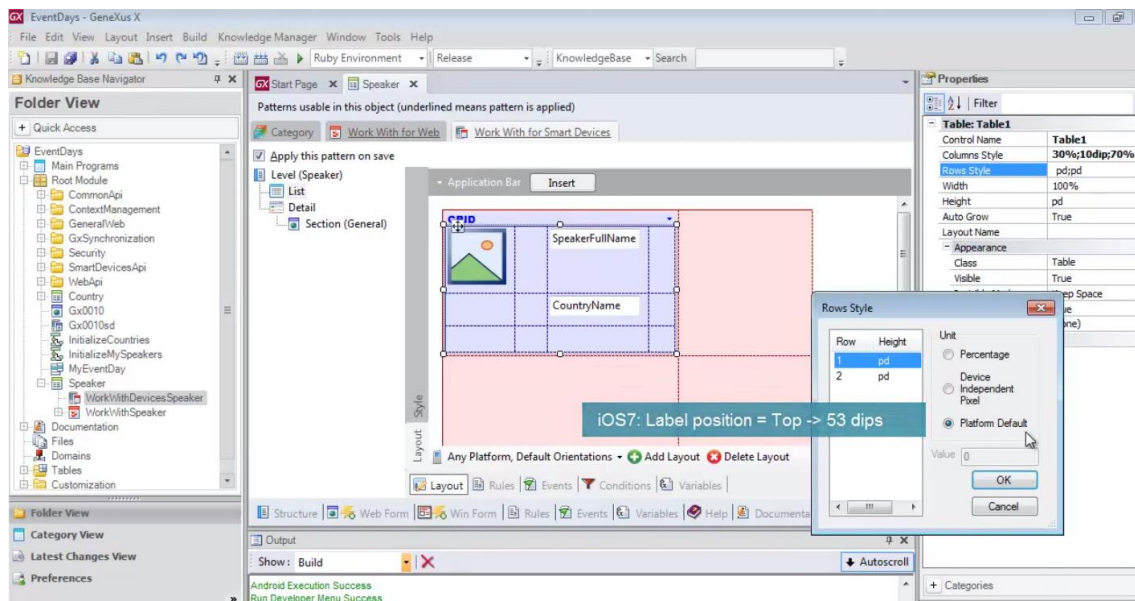
Esta unidad difiere de plataforma en plataforma. Corresponde a: **Using the best value depending on the platform and the context**



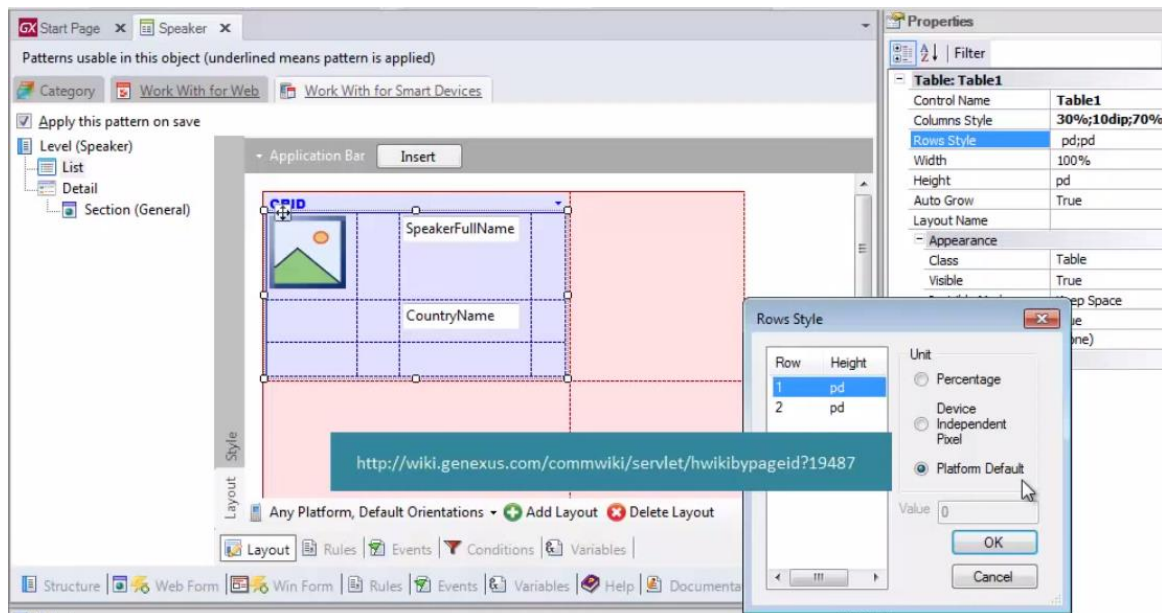
Por ejemplo para Android, con **Label Position = Top**, corresponde a 64 DIPs



Mientras que en iOS7, a 53

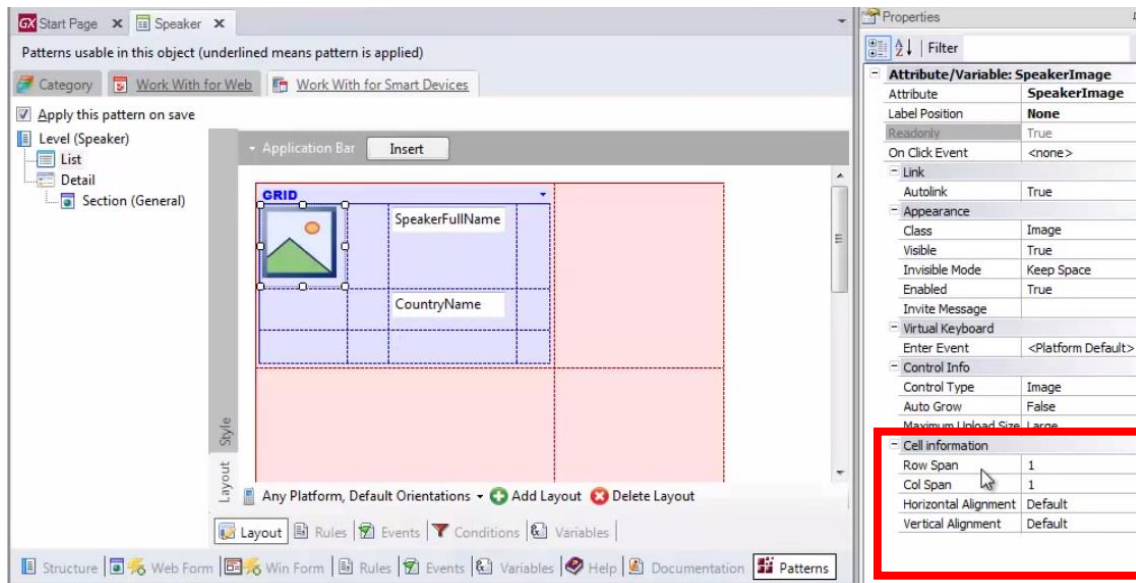


Vea todos los valores que asume, aquí:

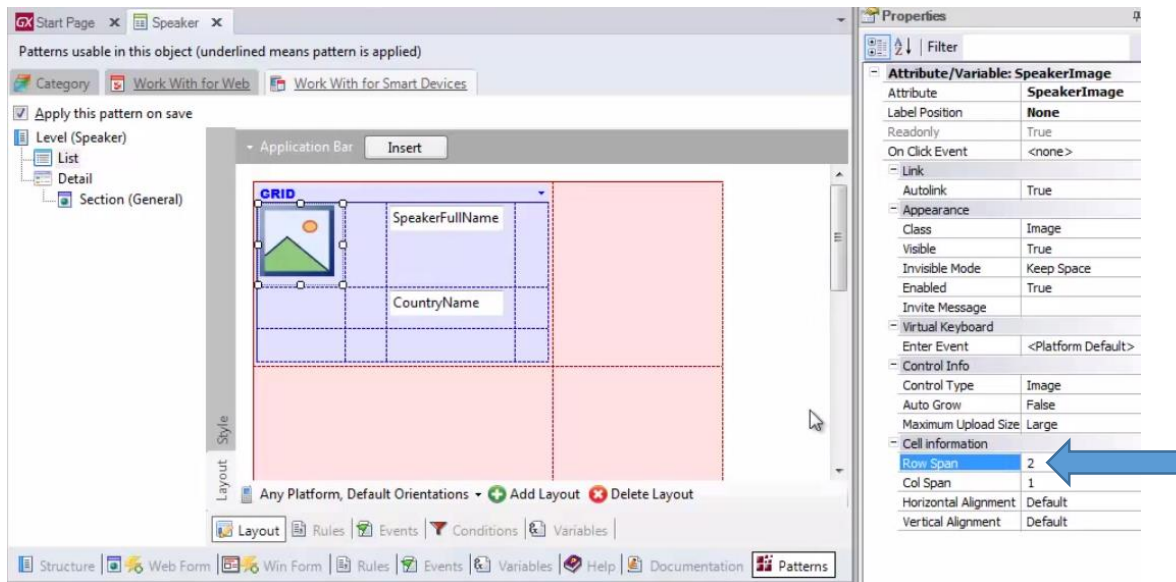


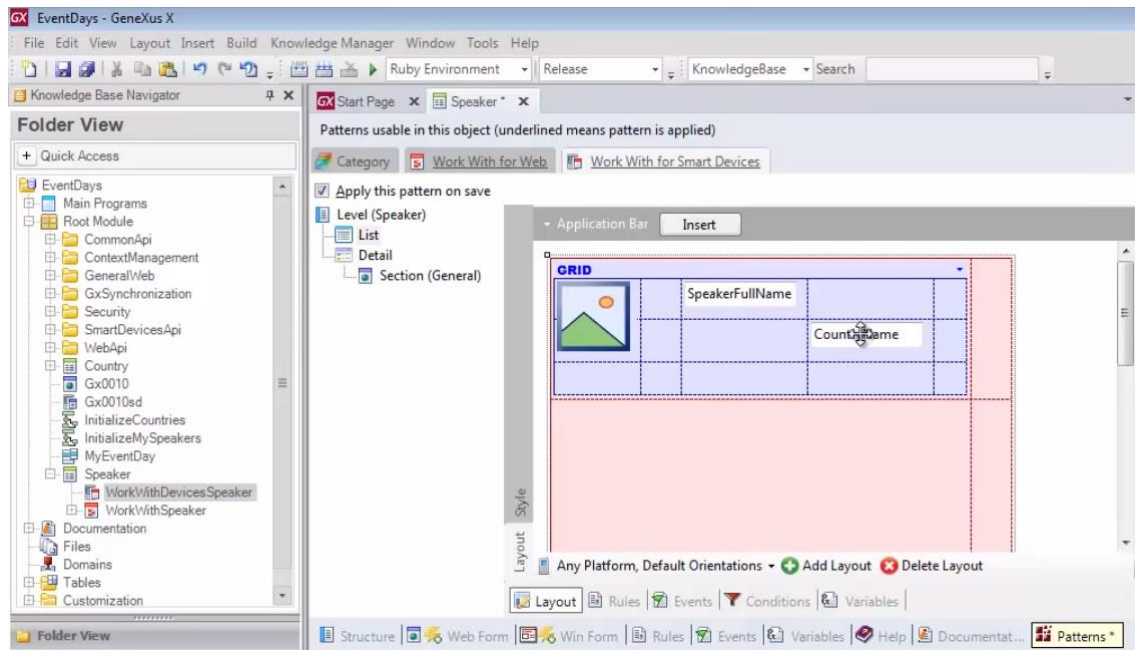
Vamos a hacer que nuestra imagen se expanda en las 2 filas.

Sobre la imagen, en **Cell Information**:

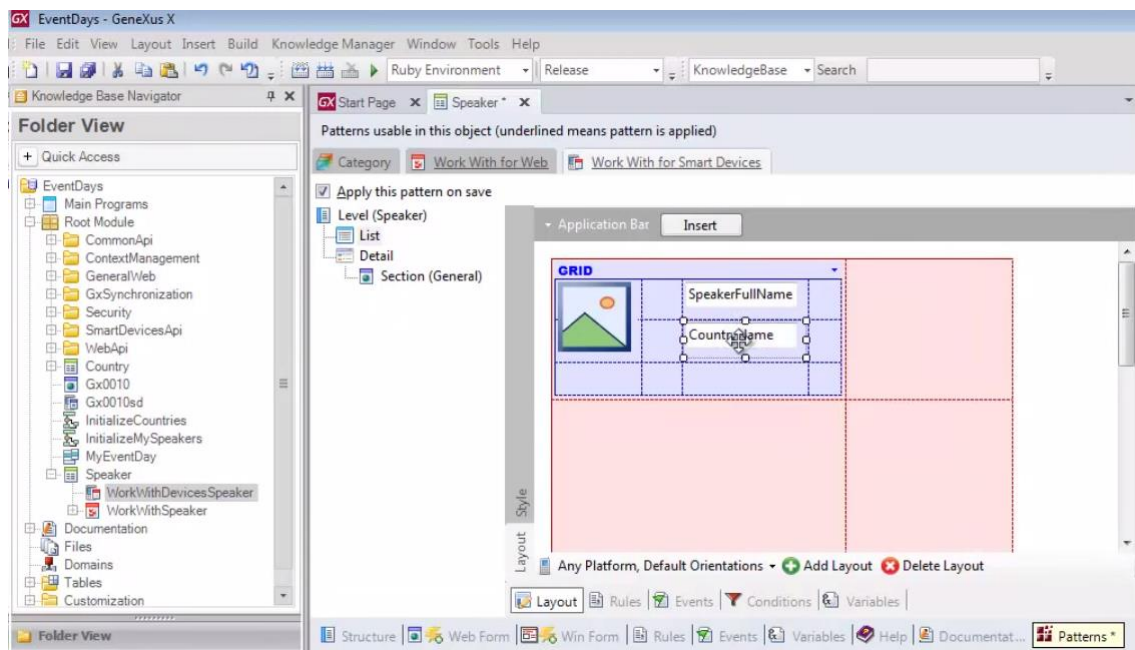


Row Span: 2

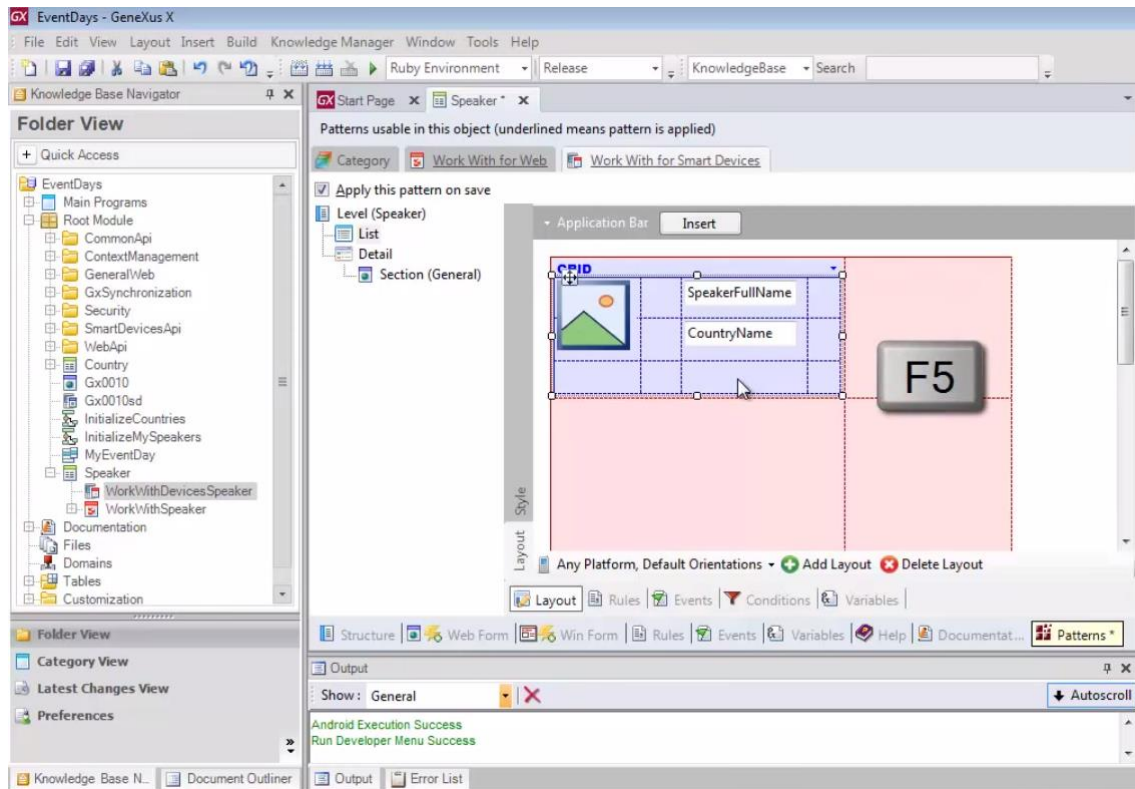


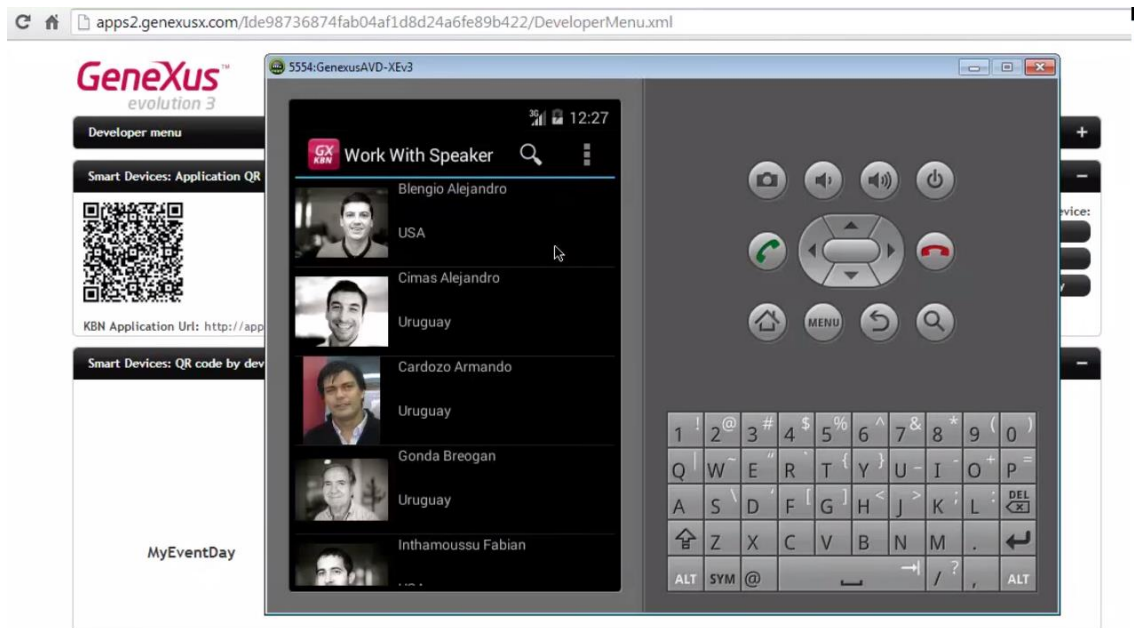


Y acomodamos los controles para que nos queden como queríamos:



F5...





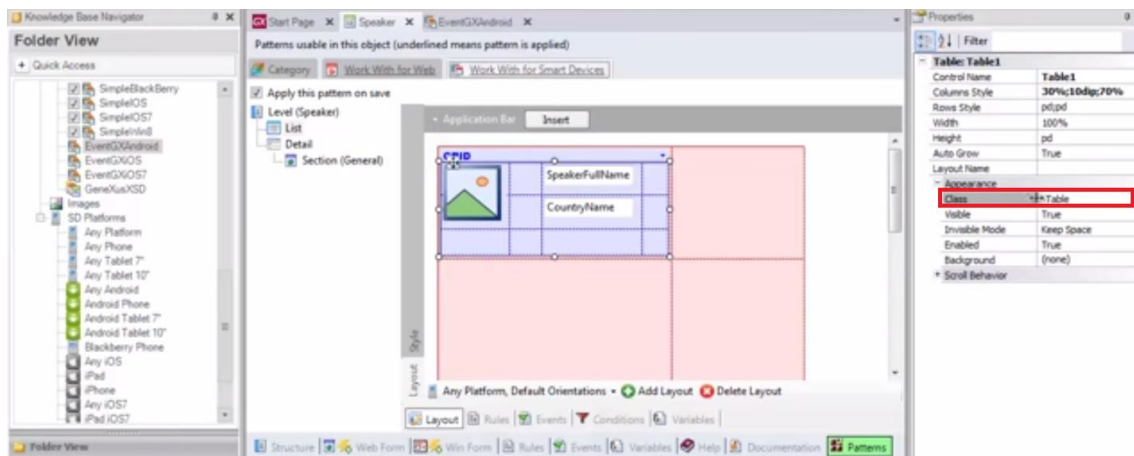
Si ahora queremos que el fondo de la tabla, en lugar de negro sea gris, y que cuando el usuario hace TAP sobre una línea



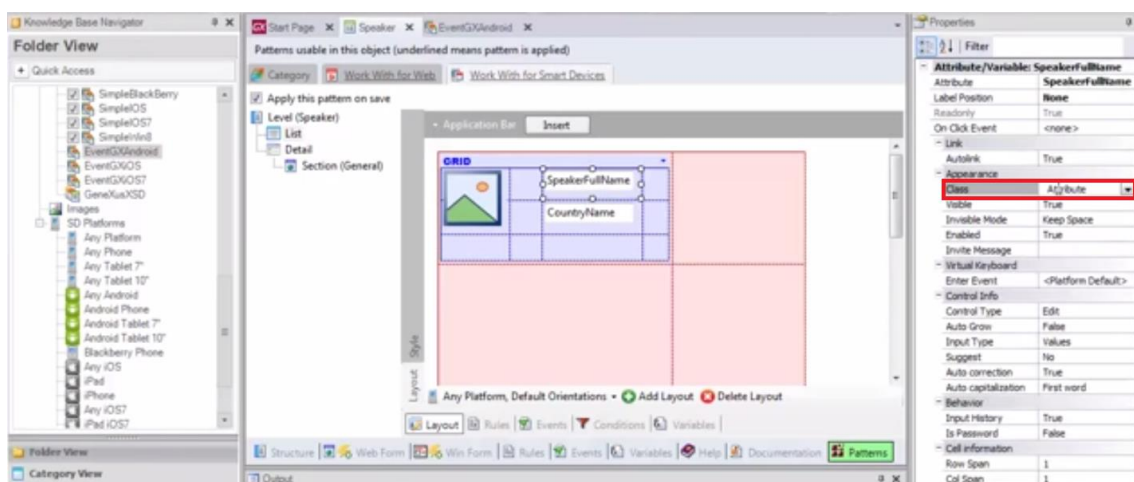
En lugar de mostrarse de este color gris, se muestre azul, y el nombre completo del speaker se muestre también en azul, ¿dónde lo configuramos?

En las clases de cada control.

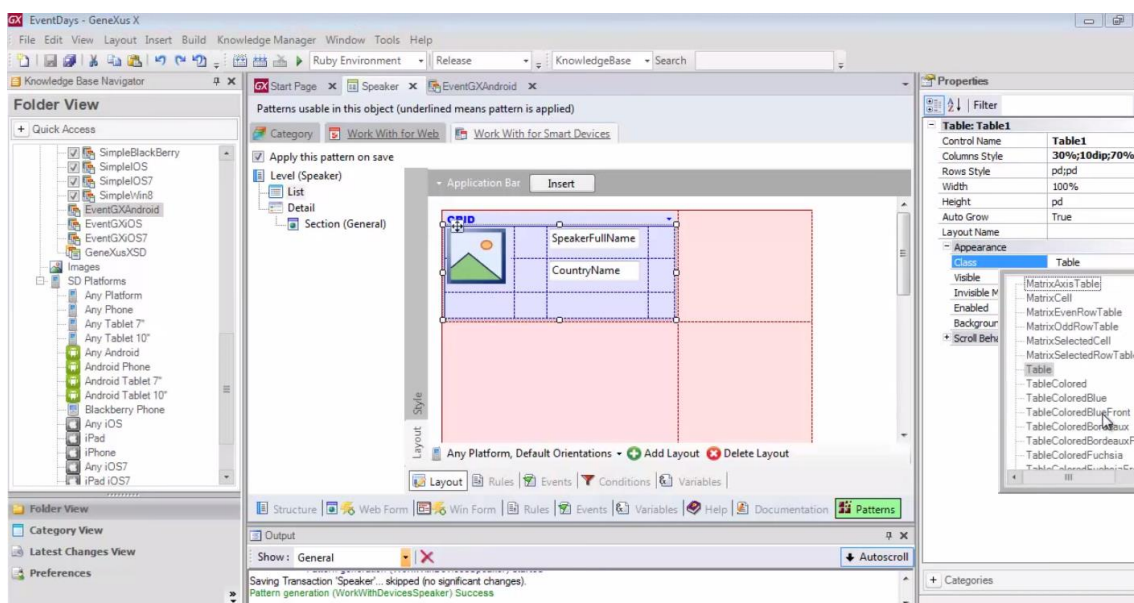
Del control tabla por un lado



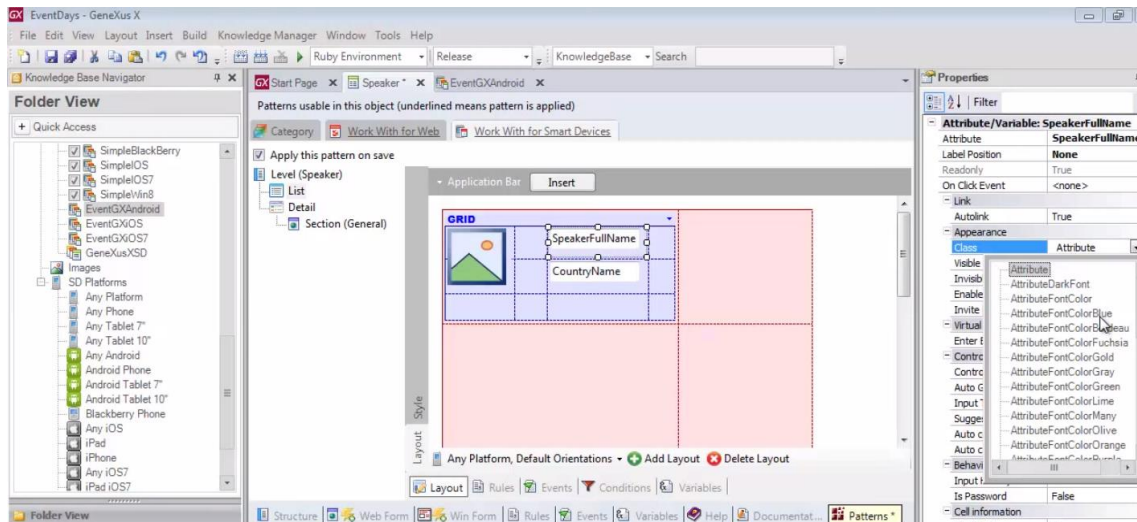
y del control atributo por otro



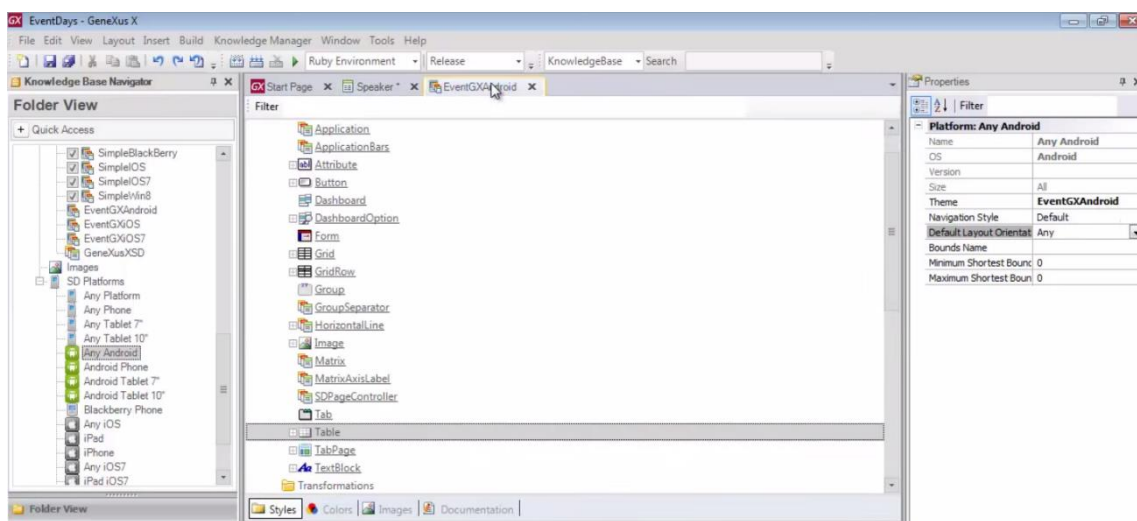
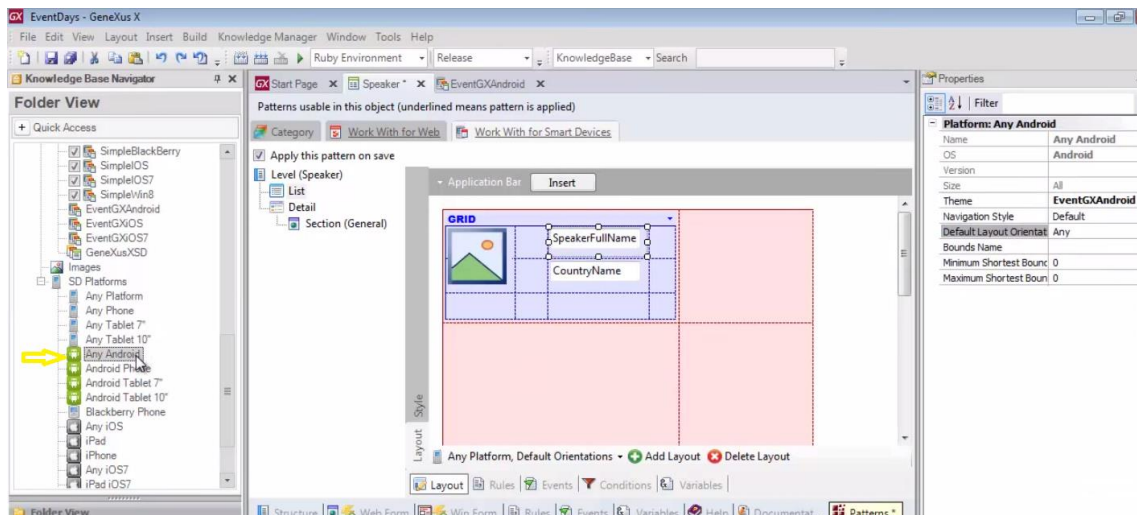
Para el control tabla entonces, vamos a cambiar su clase predefinida **Table**, por la clase: **TableColoredBlueFront**



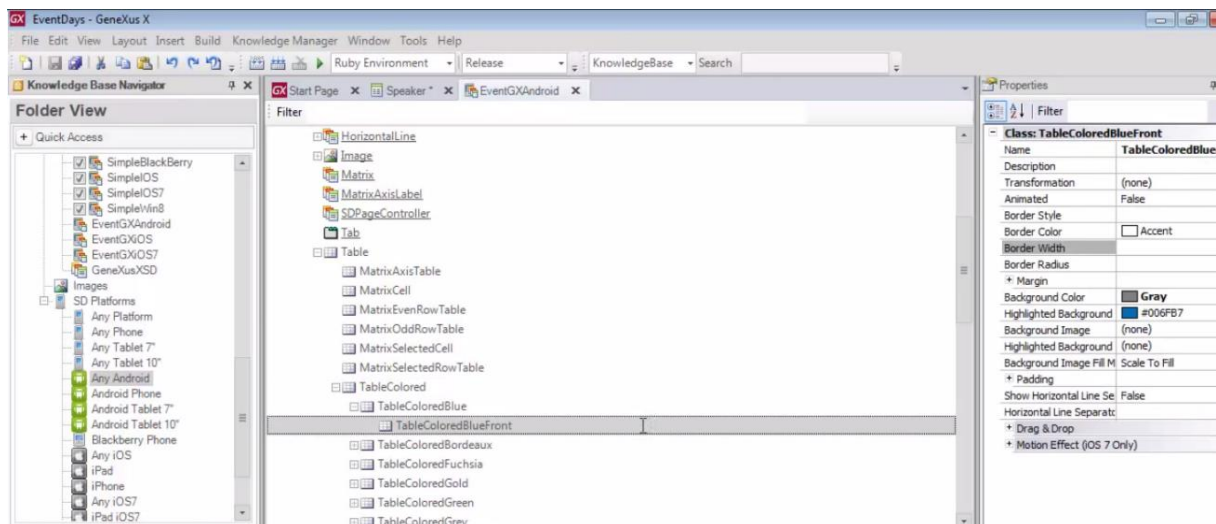
Para el caso del atributo, vamos a cambiar su clase de: Attribute por: **AttributeFrontColorBlue**



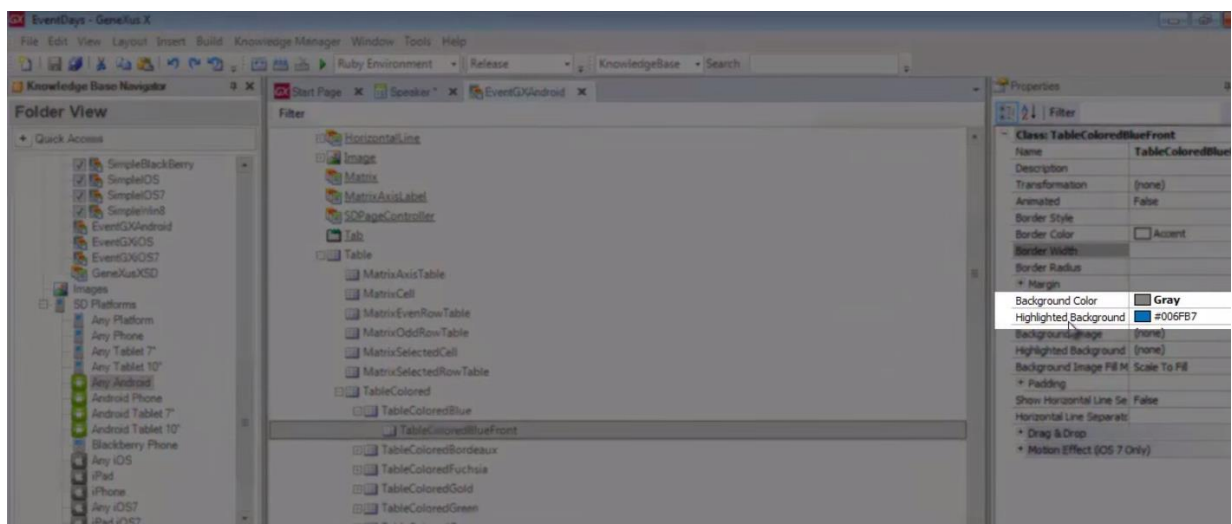
Si vamos al tema correspondiente a la plataforma Android que estamos ejecutando (recordemos que era EventGXAndroid)



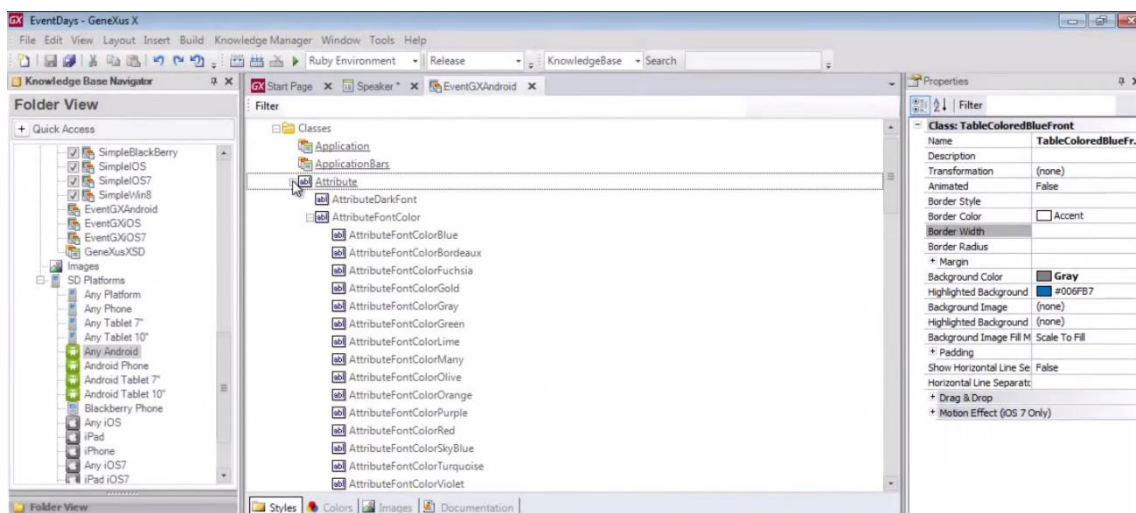
y buscamos esas clases..



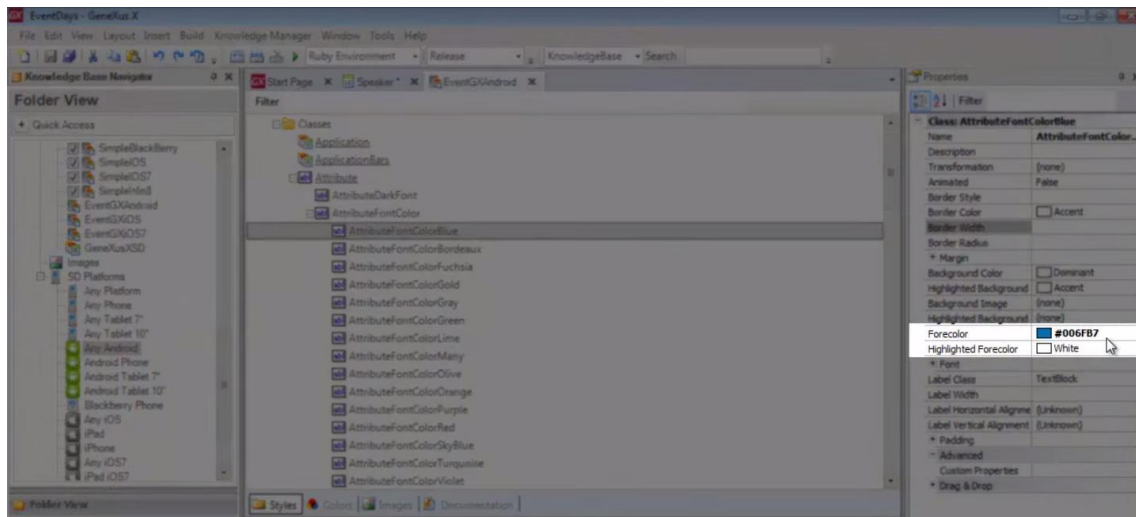
vemos que la clase **TableColoredBlueFront** es la que asume los valores que queremos para las propiedades **BackgroundColor** y **HighlightedBackground** de la tabla (por un lado):



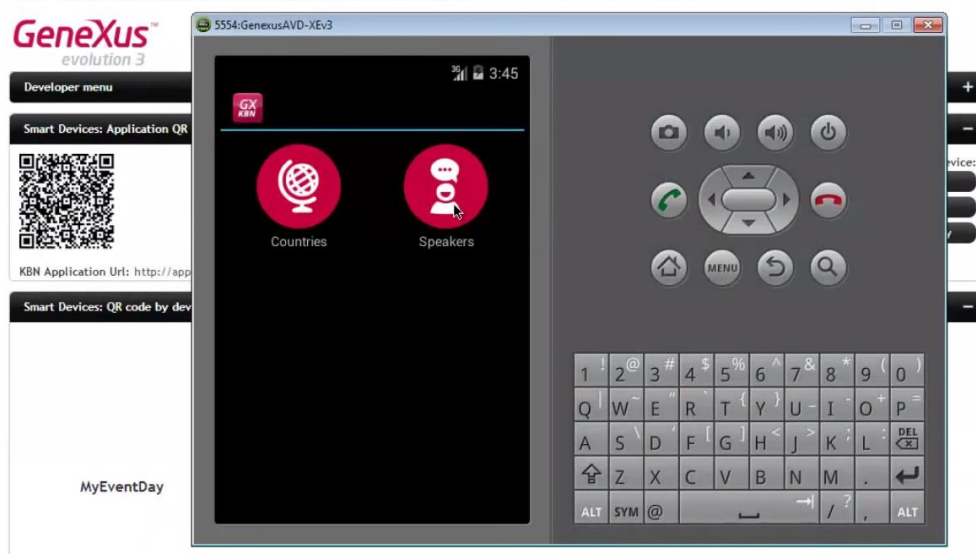
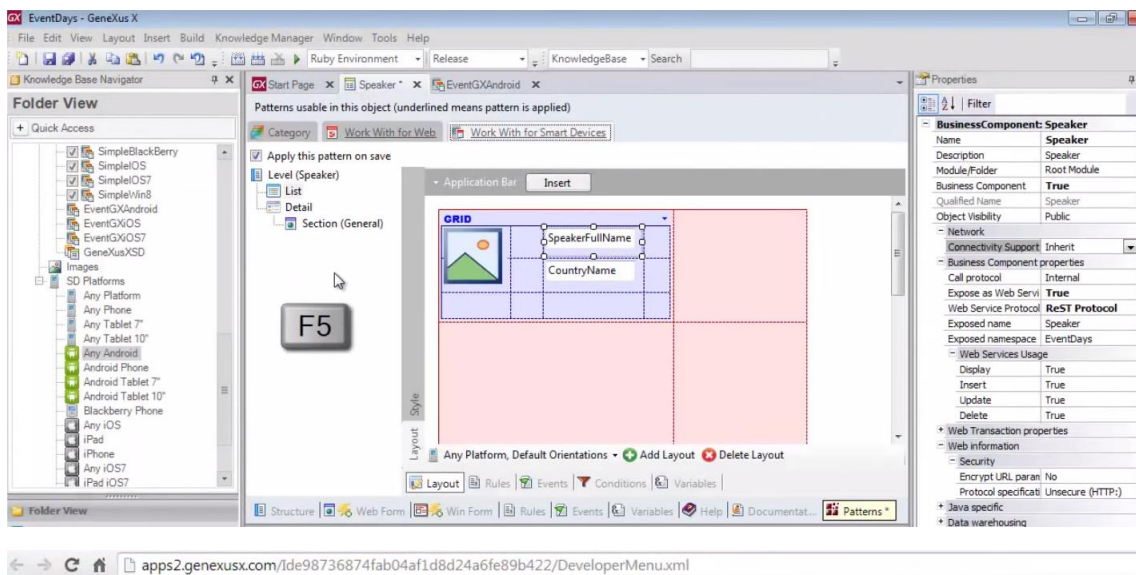
Y para el atributo



estas dos:

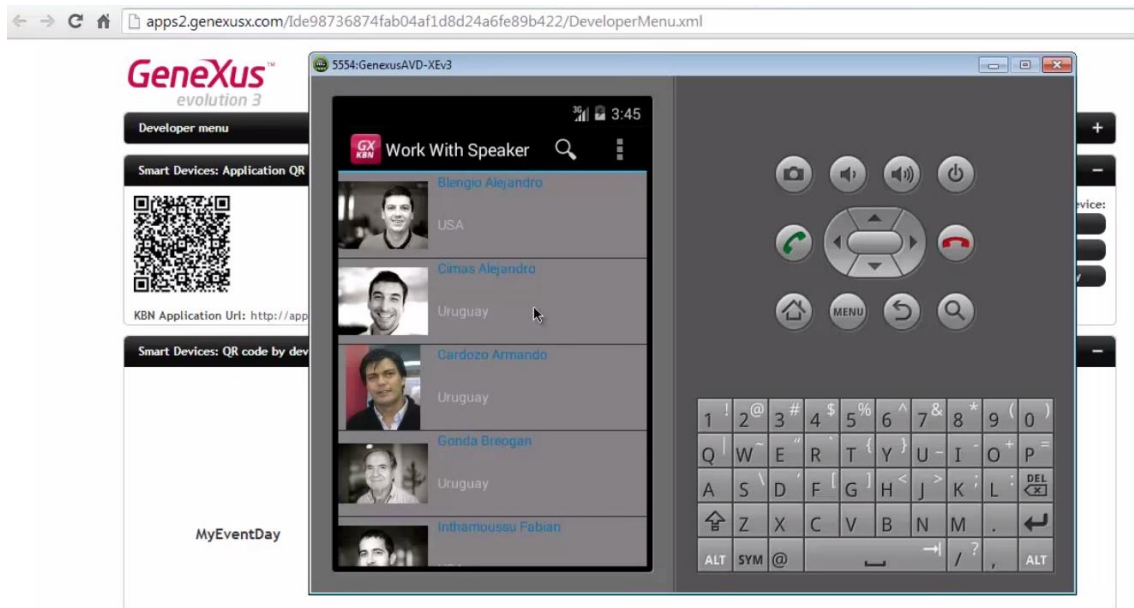


F5



Y aquí lo vemos:

Video filmado con GeneXus X Evolution 3



Si hacemos TAP sobre un elemento, vemos como cambia el color de fondo:

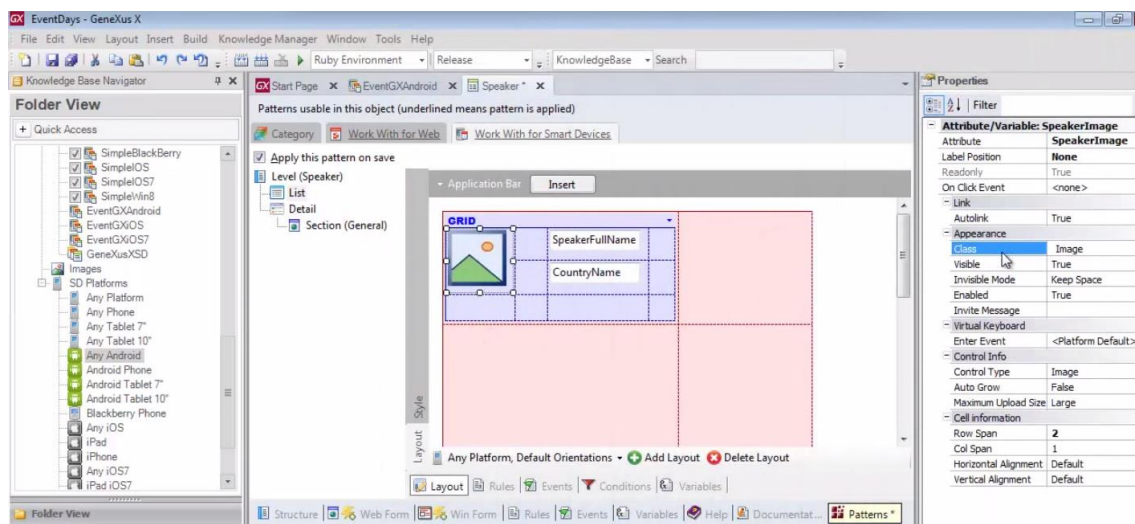


Ahora prestemos atención **a la forma** en la que se muestra cada imagen dentro del espacio de la celda en la que está contenida

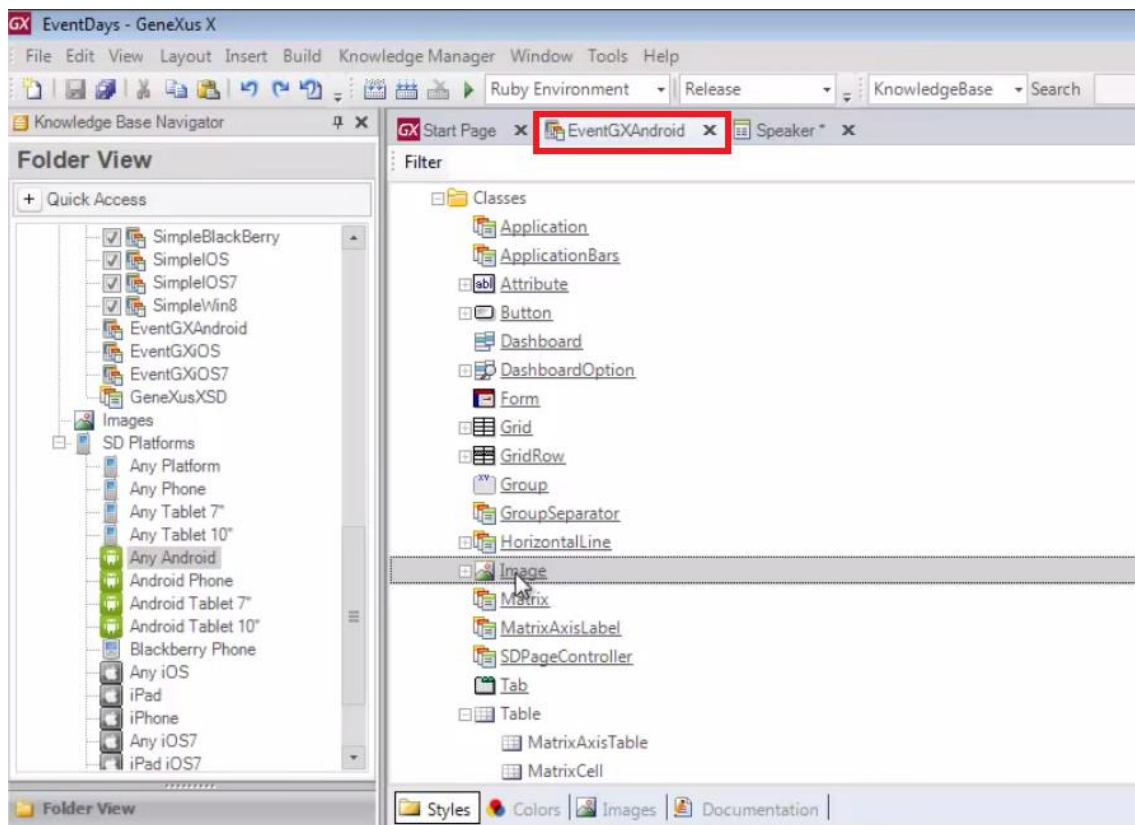


¿Se están escalando estas imágenes? Y de ser así: ¿cómo?

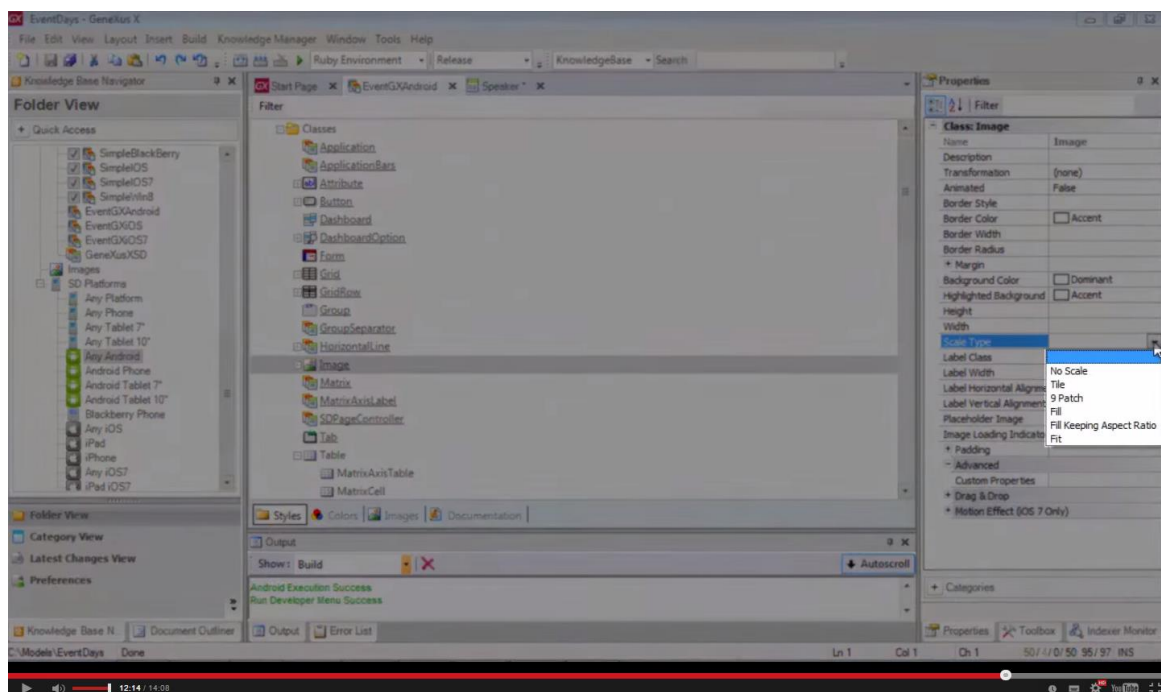
Observemos para el control imagen, su clase: Image



Si vamos a las propiedades de esta clase, dentro del tema

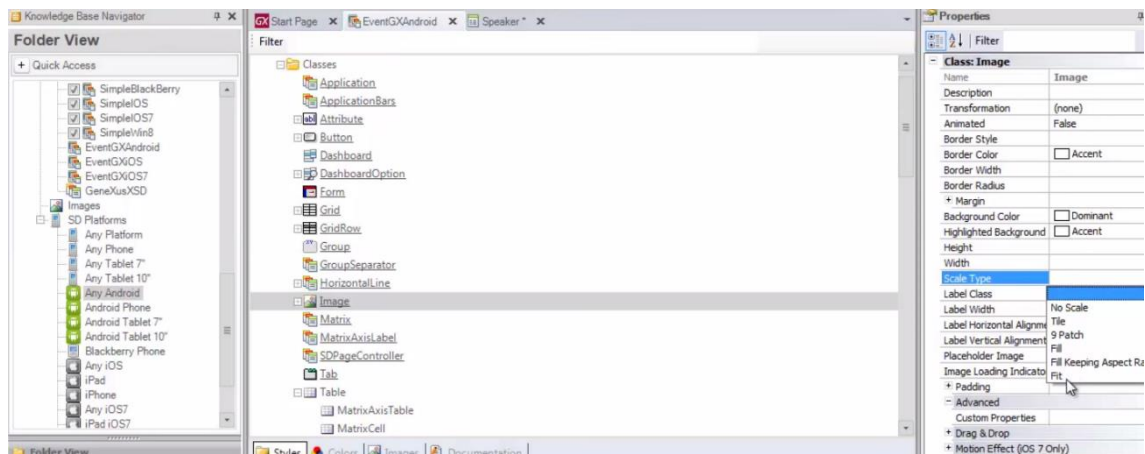


Encontramos que existe esta propiedad **Scale Type**, que puede asumir uno de los siguientes valores



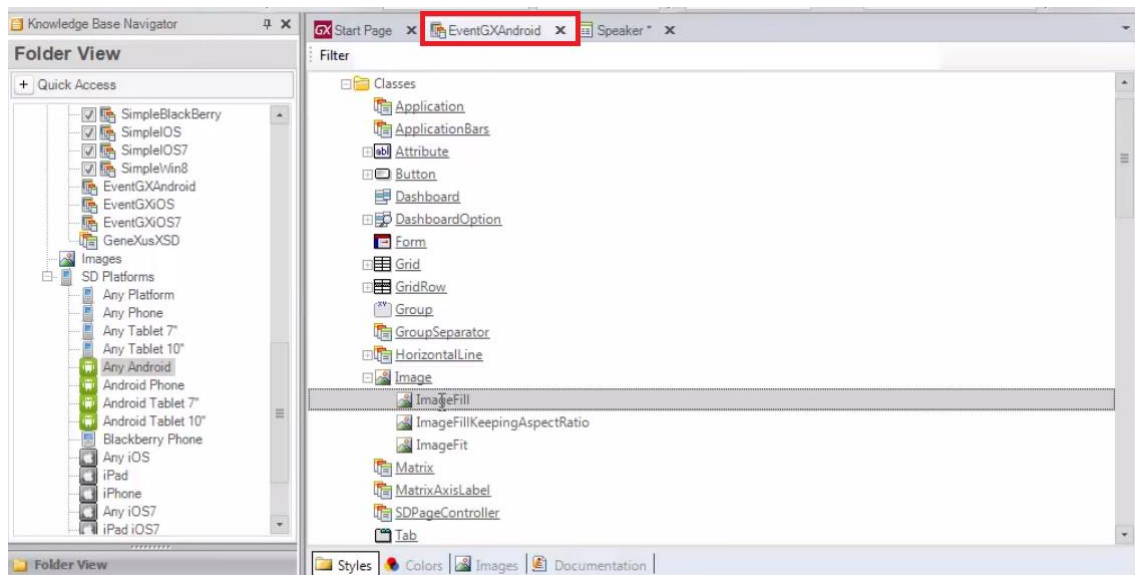
Estos valores van a determinar la manera en que se escala la imagen, si es que se escala, dentro del área del control en el que está contenido.

Si no indicamos nada, el valor asumido es: Fit

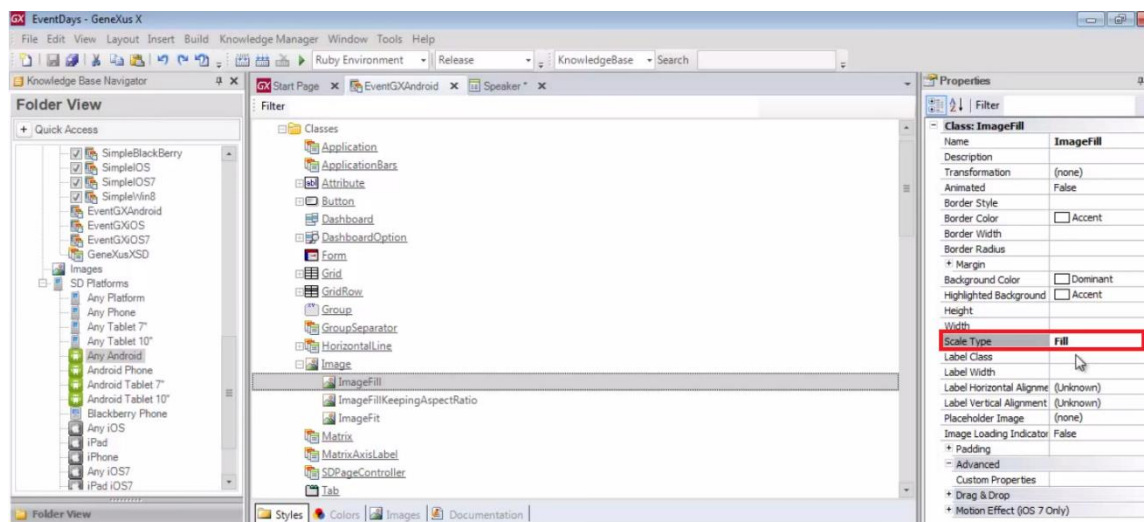


Lo que va a provocar que la imagen se escale en ancho y largo, de manera de poder entrar en el espacio manteniendo su aspecto original.

Recordemos que en este tema, se han agregado clases

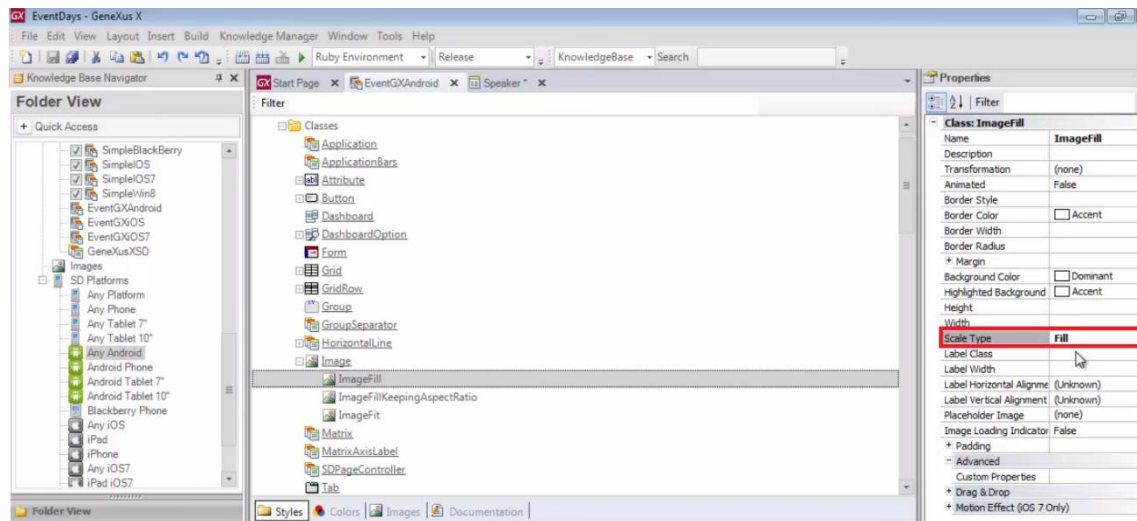


Que únicamente configuran esta propiedad

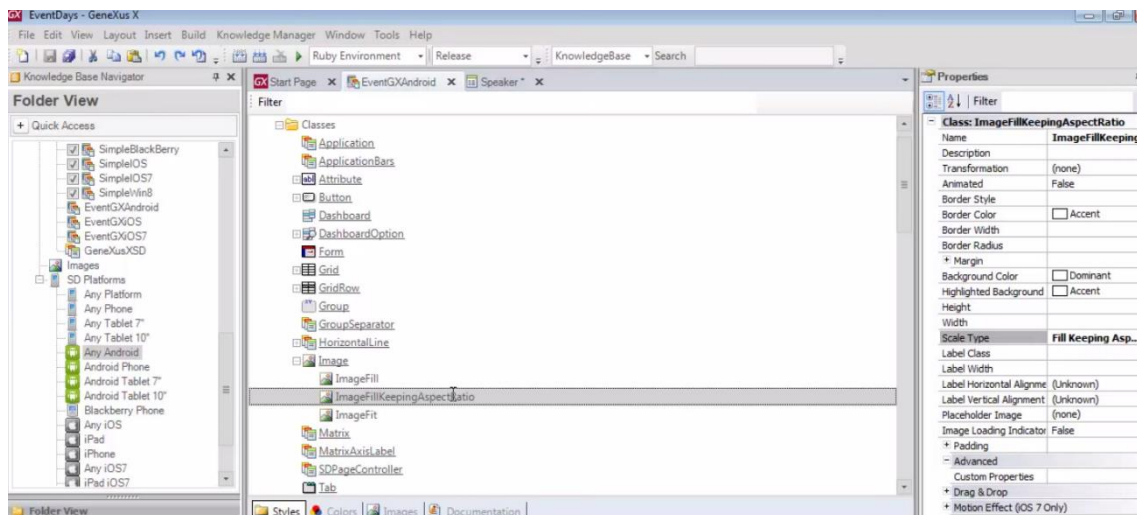


para adoptar alguno de los otros valores.

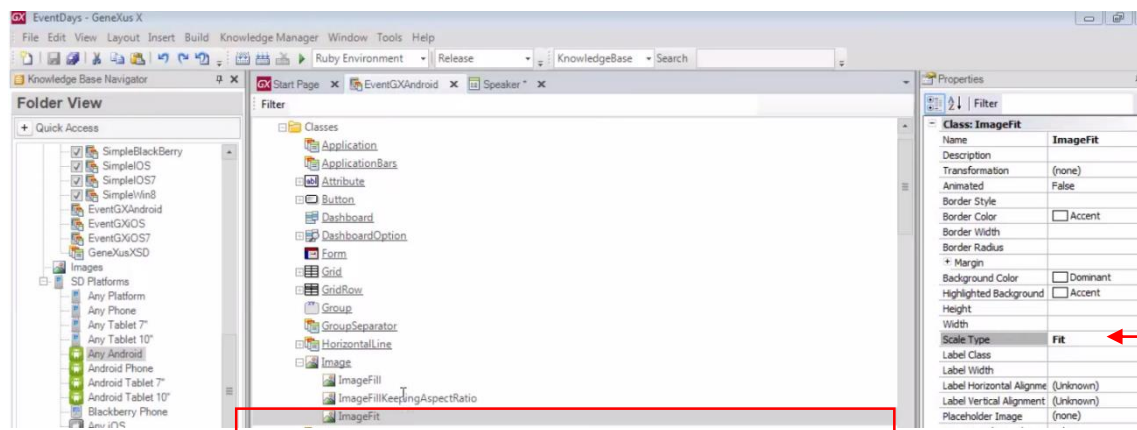
Fill:



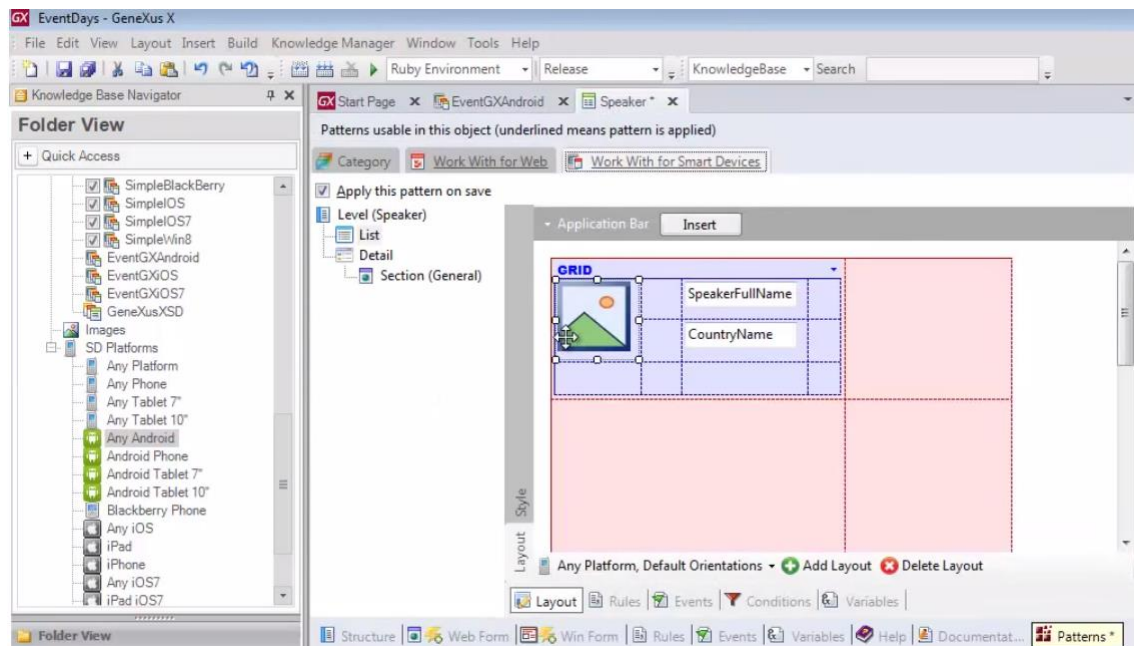
Fill Keeping Aspect Ratio:



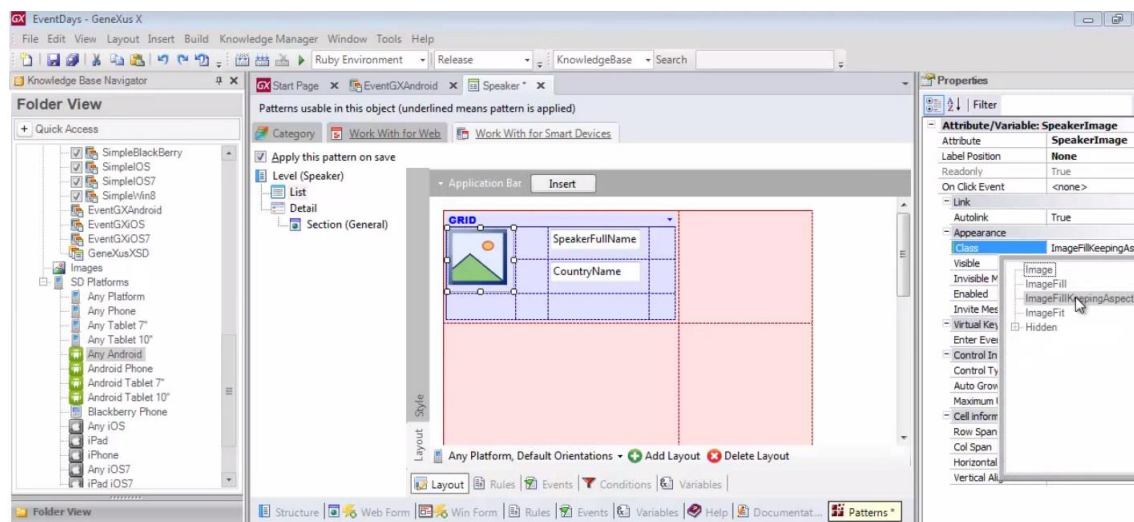
O Fit:



Supongamos que queremos que nuestra imagen se haga más grande o más pequeña, de manera tal de poder ocupar el espacio completo de la celda

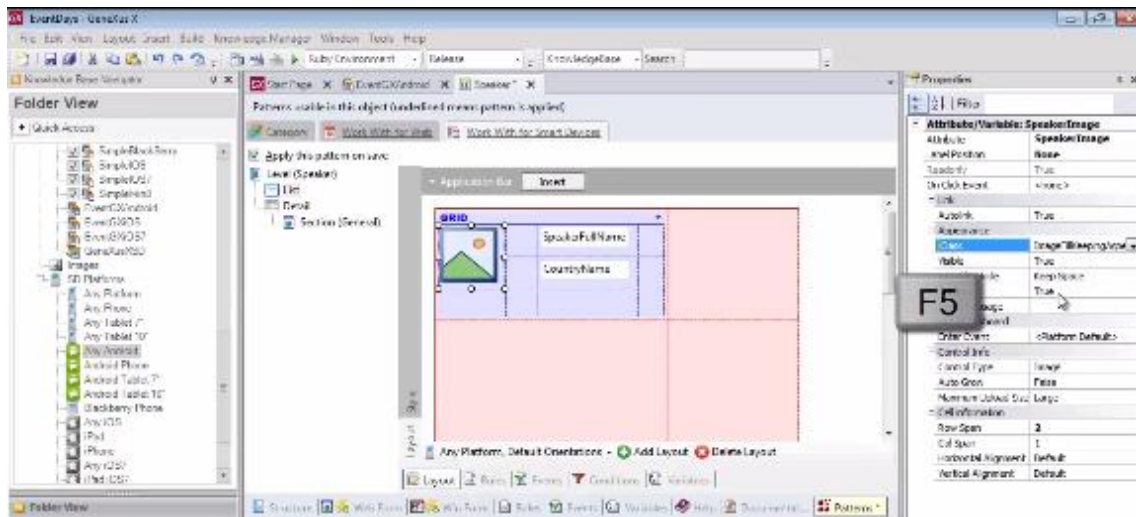


Para ello vamos a cambiar la clase

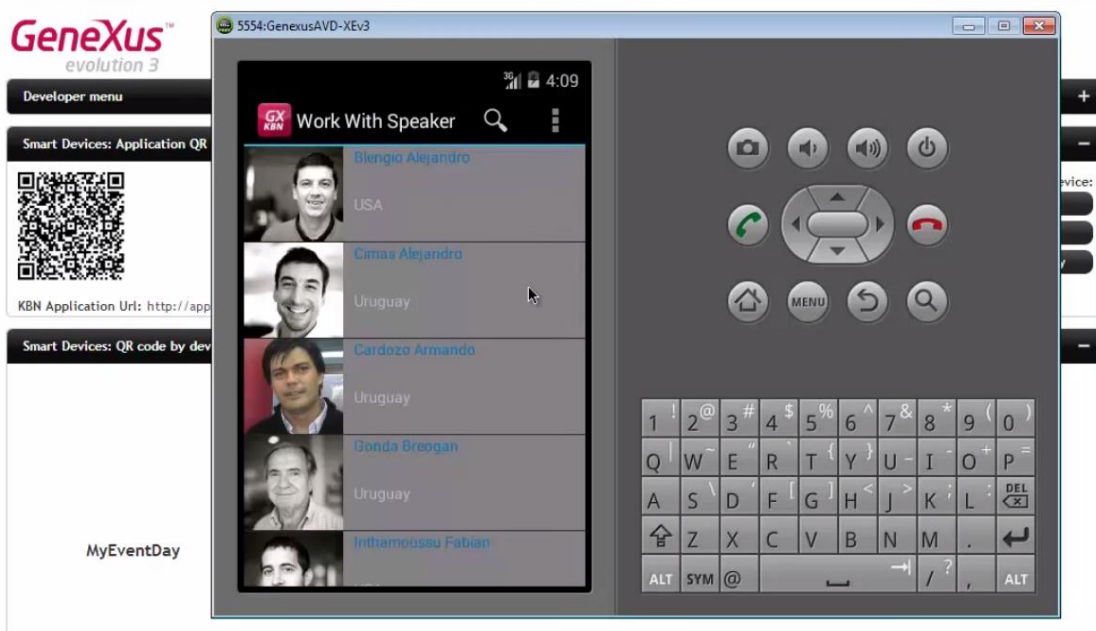


A ImageFillKeepingAspectRatio

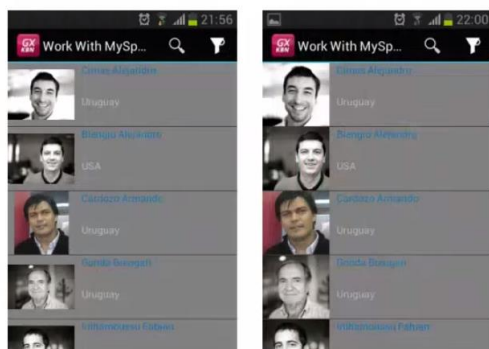
Veámoslo en ejecución.. F5



apps2.genexus.com/Idc98736874fab04af1d8d24a6fe89b422/DeveloperMenu.xml



Controls in layouts: image



No Scale	Respects the original size of the image, independently of the control area size.
Tile	The image is not scaled. It is repeated horizontally and vertically to fill the control size.
9 Patch	The image must have the Scalable Image Property set to true. This images contain information about how they should be scaled.
Fill	The image is scaled in width and height in order to fill the whole size of the control area.
Fill Keeping Aspect Ratio	The image makes bigger or smaller in width and height in order to fill the whole size of the control area, but keeping the aspect of the image. For example, if the image size is 100x200, and the control size is 50 x 50, then the image size is converted to 50 x 100.
Fit	The image scales in width and height in order to see it at all, and keeping the aspect of the image. For example, if the image is 100x200, and the control is 50 x 50, then the image is converted to 25 x 50. This is the default value.

Aquí vemos un resumen de lo que se consigue con cada valor.

Encontrará más información sobre todo esto en nuestro wiki.

Controls in layouts

Control Labels, tables, images

Grid: Multiple layouts per row

Control types

Detail <section content>

Canvas & Transformations

Con esto terminamos el primer punto que nos habíamos planteado relativo a los controles en los layouts.. y ahora pasaremos a estudiar el segundo

