

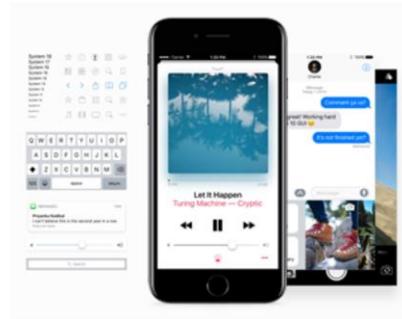
Smart Devices

GeneXus™ 15

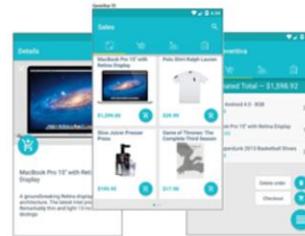
Design guides & GeneXus

Design Guides

iOS
(Human Interface guidelines)



Material Design for Android (5 +)



Como sabemos uno de los motivos por los que necesitamos que las aplicaciones sean nativas es que conserven el look & feel de todas las demás apps del dispositivo del usuario.

Cada plataforma define guías de diseño para sus aplicaciones, tanto en lo que hace a la User Interface como a la User eXperience.

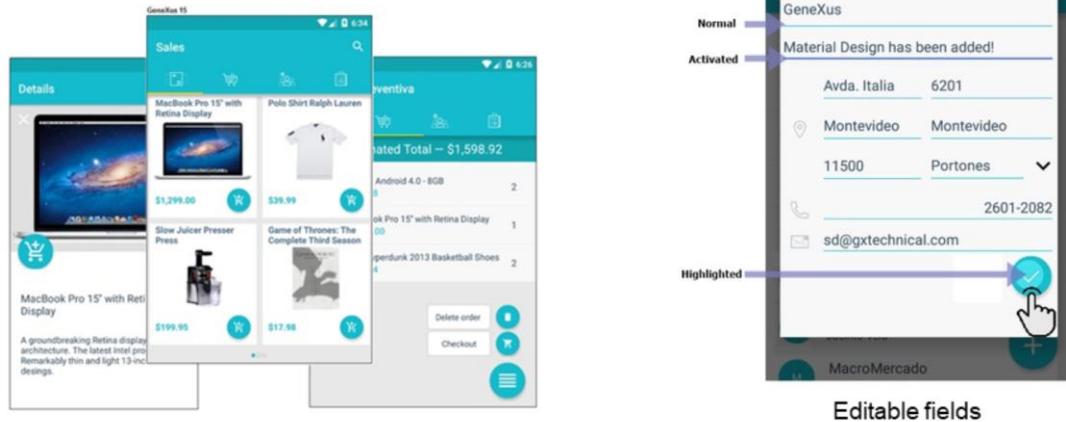
Desde Android 5 se cuenta con las guías conocidas como Material Design, y para iOS son las publicadas en el sitio de Apple (se conocen como HIG: Human Interface Guidelines).

Design Guides: Android

Veamos las de Android.

Material Design for Android (5 +)

1. General appearance



La plataforma Android, desde la versión 5 ha incorporado las guías de diseño conocidas como Material Design, para crear un ecosistema particular y característico. Esto permite a los desarrolladores customizar el look & feel de sus aplicaciones, haciéndolas más user-friendly con una UX optimizada.

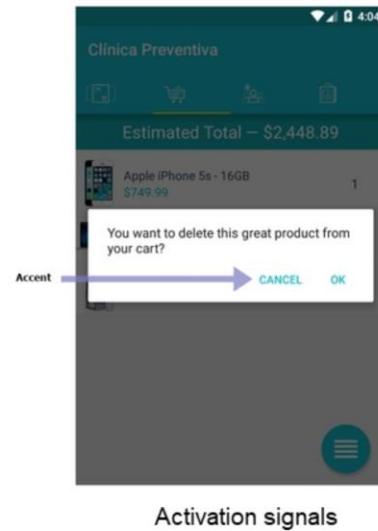
Como veremos, estas funcionalidades serán ofrecidas por GeneXus de dos maneras: a través de **propiedades personalizables**, y también serán provistas por **default**.

Aquí mostramos una aplicación de ventas desarrollada con GeneXus 15, que sigue los lineamientos de Material Design, donde la **uniformidad en el uso de los colores para determinados elementos comunes de toda app** se torna fundamental.

Por ejemplo, a la hora de editar campos, vemos que se subrayan con un color uniforme los que no están activos en un momento dado, y con otro el campo con el que se está trabajando. Asimismo al hacer tap sobre un botón, se muestra resaltado siempre con el mismo color, en toda pantalla que haga lo mismo.

Material Design for Android (5 +)

1. General appearance



La Action Bar es uno de los elementos más importantes de toda aplicación. Su color se convierte en el "primary color" de la aplicación, que distingue a la marca. Las guías del Material Design indican que el color que debe asumir la Status Bar, es decir, la barra de arriba del todo (la que contiene la hora, indicadores de batería, wifi, notificaciones, etc.) debe tener un color 700 tints más oscuro que el primario.

Además, el color de los íconos embebidos en al application bar deben ser más claros que el "primary color", armónicos con éste, y debe ser siempre el mismo cuando se navega por distintas pantallas de la app.

También se establece de manera uniforme en toda la aplicación el color de los controles que son activados por el usuario (ej: tap sobre un botón o un texto para confirmar en una pantalla modal, radio buttons, checkboxes, datepicker, etc.).

Material Design for Android (5 +)

2. Status Bar color



3. Elevation

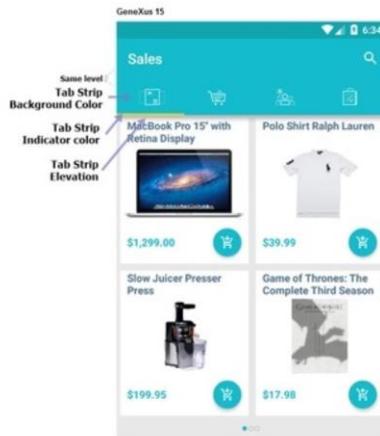


La status bar podría ser escondida, es decir, vuelta transparente, bajo ciertas condiciones, por ejemplo, cuando el usuario desliza hacia arriba la pantalla para ver lo que hay más abajo.

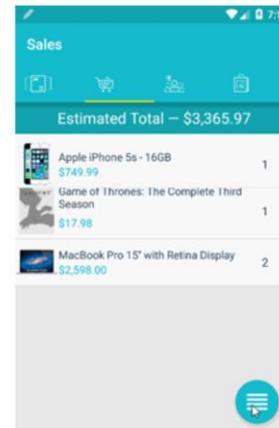
Por otro lado, una sombra bajo el control lo destaca, indicándole al usuario que es un control seleccionable. Por eso se le dice "elevation" a la propiedad que permite configurar este efecto de altura.

Material Design for Android (5 +)

4. Tab strip



5. Touch ripples



Las pestañas se muestran con una imagen y un color de fondo, un indicador del tab activo con un indicador de color y con cierta elevación.

Touch ripples: es utilizado para notificar visualmente al usuario que el tap que ha realizado ha tenido efecto. Cuando el usuario final hace tap sobre un control que tiene un evento asociado, un círculo destacado se expande desde el punto donde hizo el tap hasta los bordes del control, rellenándolo por completo, mostrando que el evento se está disparando. Esto es provisto automáticamente por GeneXus.

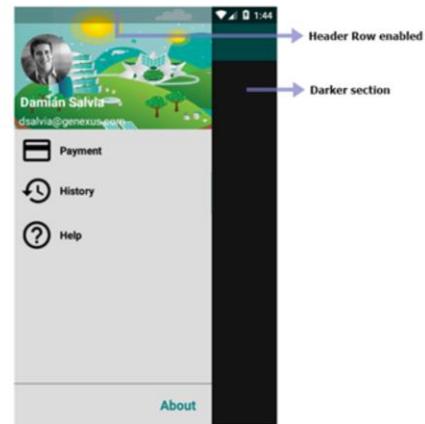
Material Design for Android (5 +)

6. Task color



7. Slide menu

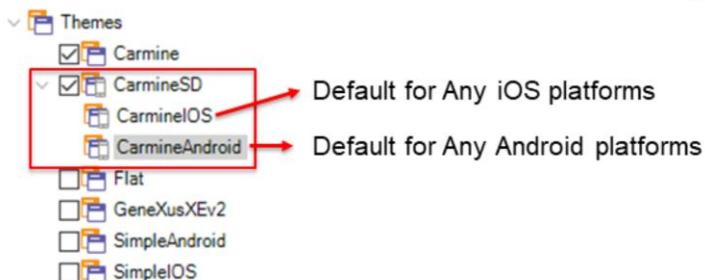
Provided by default
by GeneXus



Los dispositivos Android suelen proveer tres botones físicos/"capacitivos", cada uno dedicado a una acción particular. Además de los botones de back y home, tenemos el botón que muestra con un tab switcher una cascada de imágenes en miniatura de las apps recientemente cerradas por el usuario, para poder volver a ellas. Pueden aparecer como cascada, o como una especie de grid horizontal. De estas imágenes en miniatura se muestra su application bar, sus íconos y etiqueta. Es decir, ahora se muestran como se ven en la aplicación. Si bien las aplicaciones Android empiezan a incluir estos botones no como botones físicos sino como botones provistos por software, la idea es la misma.

Cuando el estilo de navegación es Slide, es decir, cuando el menú principal se despliega como una ventana desde la izquierda, su tamaño tiene que seguir las guidelines, y dejar una sombra en la sección derecha cuando está despegada. Además, si tiene Hero image (es decir, la imagen que aparece arriba del todo ocupando todo el ancho), obsérvese que la ventana del menú debe alcanzar la status bar, preservando su opacidad. Esto es provisto por default por GeneXus.

New Default themes for SD



New default for Android Base Style in Main SD object

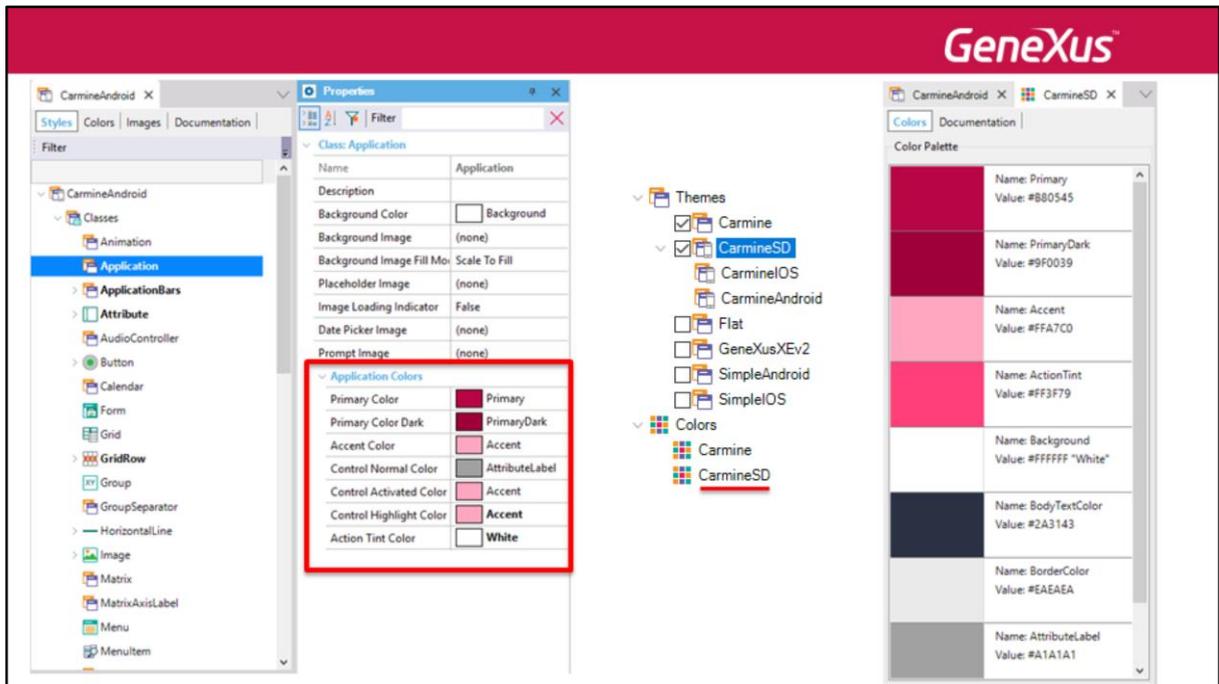
Android	
Android Version Code	1.0
Android Version Name	1.0
Update URL	
Android Package Name	com.artech.upg15.eventday
Android Application Icon	(none)
Android Notification Icon	(none)
Android Portrait Launch Image	(none)
Android Landscape Launch Image	(none)
Android Base Style	Light with Dark Action Bar
Android Maps API	Google Maps API v2

Nuestras aplicaciones deben respetar las guías de diseño para Android y para iOS. Gran parte del comportamiento ya se da por default sin que el desarrollador deba hacer nada. Pero otra parte del comportamiento veremos que se implementa mayormente en los themes y otra parte como propiedades de los controles.

A partir del upgrade 6 de GeneXus 15 hay un nuevo theme, CarmineSD, con dos sub-themes, CarmineiOS y CarmineAndroid que serán los defaults de estas plataformas.

Además para Android se ha modificado el default de la propiedad Android Base Style pasando a "Light with Dark Action Bar". Esta propiedad es del objeto main de la app SD.

[Demo] lo que hará que la aplicación se vea con un fondo claro.



Decíamos que nuestras aplicaciones deben respetar las guías de diseño de Android y de iOS.

Si vamos al theme default para Android, vemos que en la clase **Application** aparecen un conjunto de propiedades agrupadas bajo el nombre Application Colors, que serán las que permitirán definir los colores de acuerdo a los criterios de Material Design que vimos.

¿Qué colores son los definidos como Primary, PrimaryDark, Accent, etc? Los que vienen por defecto en la paleta Colors / CarmineSD.

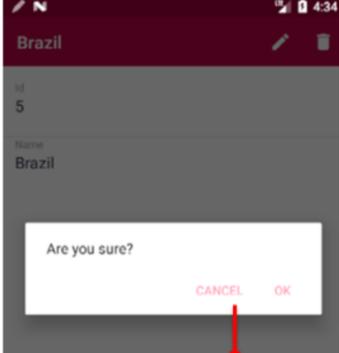
GeneXus

Themes

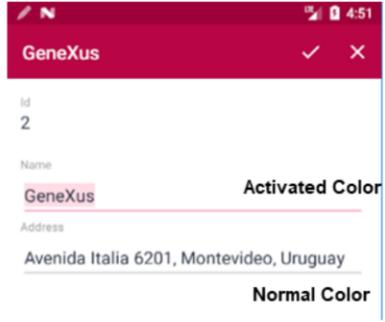
Application Colors	
Primary Color	Primary
Primary Color Dark	PrimaryDark
Accent Color	Accent
Control Normal Color	AttributeLabel
Control Activated Color	Accent
Control Highlight Color	Accent
Action Tint Color	White



Primary Color Dark Primary Color Action Tint Color



Accent Color



Activated Color

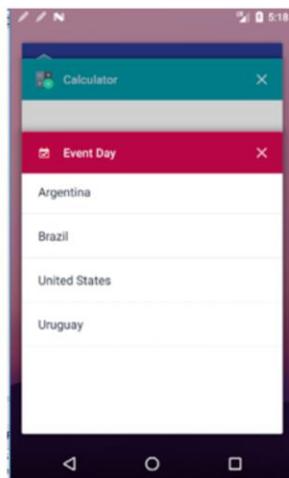
Normal Color

Acá vemos cómo esos colores están siendo utilizados para respetar el Material Design en la app generada.

[Demo] Este es el accent color... Primary Color Dark... Primary Color... Action Tint Color... Activated Color... Y Normal Color.

More Material Design

Default behavior



App switch color



Touch ripples

GeneXus también provee por default dos features que ya mencionamos de Material Design:

La primera es que el app switch color coincide con el brand color de la app.

La segunda es la conocida como el efecto Touch Ripples, que produce un tipo de onda expansiva cuando el usuario hace tap sobre el control, como ya dijimos. En el ejemplo, sobre la fila Uruguay, para poder ver su detalle.

More Material Design

Classes: AppBar, Attribute, Button, Grid, GridRow, Group, Image, Tab, Table, TextBlock



Elevation property! (Android only)

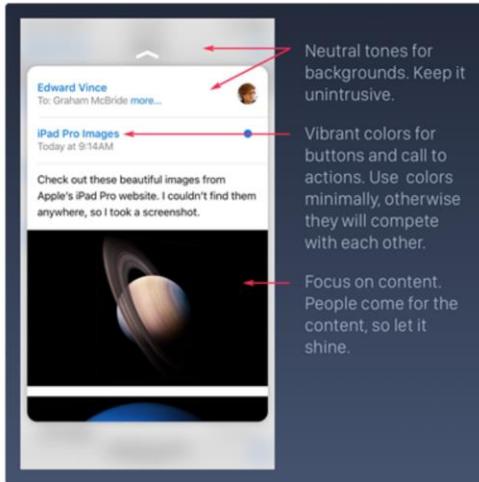
Value	by default	5	10	20
Effect				

Podemos especificar que controles de la app Android asociados a las clases que se señalan arriba, o subclases de estas clases, puedan tener un sombreado, de acuerdo al valor de la **propiedad Elevation**, de esas clases. De este modo se estarán siguiendo las guías del Material Design.

Luego vamos a mencionar cómo se combinan clases de los themes con otras propiedades a nivel de los controles para lograr efectos como el paralaje, motion en iOS, la hero image que mencionamos antes en el caso de Android, etc.

Design Guides: iOS

Veamos ahora las guías de diseño de iOS.

iOS (7 +) <https://developer.apple.com/design/>

#5AC8FA Videos	#FFCC00 Notes	#FF9500 iBooks	#FF2D55 Apple News
#007AFF Safari	#4CD964 Messages	#FF3B30 Calendar	#8E8E93 Settings
#EFEFF4 Background	#CECED2 Lines	#000000 Text	#007AFF Links

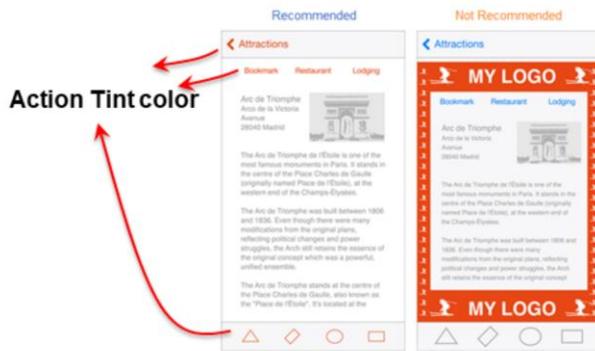
iOS uses vibrant colors to bring out the buttons. These colors tend to work well against a **white background** as much as a **black background**. Keep in mind that colors should be used sparsely, for call-to-actions and minimal branding areas like the **navigation bar**. Roughly, **only 10-20% of your design should have colors**, or they will compete too much against the content.

En el sitio de Apple se encuentran las distintas guías de diseño para las diferentes versiones de los sistemas operativos de iOS. El cambio fundamental se dio con la liberación de iOS 7.

Una de las particularidades de sus guías de diseño tiene que ver con el uso minimalista de los colores y la información de pantalla.

Observemos que sugieren colores neutrales (tonos neutrales) para los backgrounds, para que no sean intrusivos; colores vibrantes para botones y llamados a acciones... Observemos que dice que solamente del 10 al 20 % de nuestro diseño debería tener colores, para evitar competir con el contenido... Y aquí aparecen algunos de los colores sugeridos.

iOS (7 +)



Tab bar: unselected/selected



Se recomienda utilizar siempre un único color (preferentemente el de la marca) para todas las acciones que se le brindan al usuario, y no utilizar ese color para otras cosas. Así, acá vemos una pantalla con un diseño recomendado... Y otro no recomendado. De este modo el usuario sabrá que siempre que aparezca ese color, se tratará de un elemento sobre el que podrá ejecutar una acción.

Encontraremos, por otro lado, la recomendación de utilizar los íconos generales (que se muestran en la imagen) solamente (y siempre) para esas acciones, de modo de no confundir al usuario.

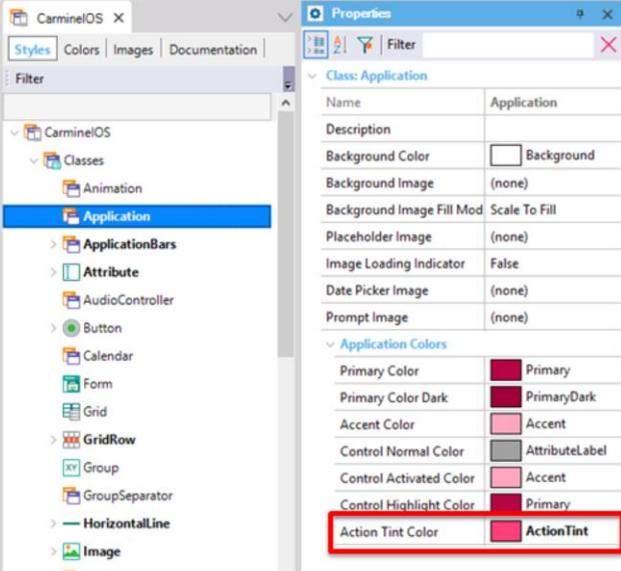
En la tab bar, que provee la navegación principal entre pantallas, se recomienda evitar el menú de hamburguesa si los ítems son pocos y agregar texto a los íconos de forma discreta cuando no son los universales. Si no están activos, se sugiere que los íconos se muestren con un contorno sin rellenar, para que reciban menos atención (deben proveerse, por tanto, dos versiones de cada ítem: para cuando está seleccionado y cuando no).

También podemos ver que hasta los tamaños que deben ocupar los controles están establecidos. Todos los íconos de la app deberían ser iguales en términos de tamaño, nivel de detalle y borde.

Themes



Theme default iOS



The screenshot shows the GeneXus IDE interface. On the left, there is a 'Themes' section with an icon of an iPhone and the text 'Theme default iOS'. In the center, there is a tree view showing the project structure for 'CarmineIOS', with 'Application' selected. On the right, the 'Properties' window is open, showing the 'Class: Application' properties. The 'Action Tint Color' property is highlighted with a red box.

Class: Application	
Name	Application
Description	
Background Color	Background
Background Image	(none)
Background Image Fill Mod	Scale To Fill
Placeholder Image	(none)
Image Loading Indicator	False
Date Picker Image	(none)
Prompt Image	(none)
Application Colors	
Primary Color	Primary
Primary Color Dark	PrimaryDark
Accent Color	Accent
Control Normal Color	AttributeLabel
Control Activated Color	Accent
Control Highlight Color	Primary
Action Tint Color	ActionTint

A nivel del theme iOS, la única propiedad de todas estas que va a tener efecto es esta última Action Tint Color, que se va a utilizar para pintar íconos y los controles estándar.

Themes

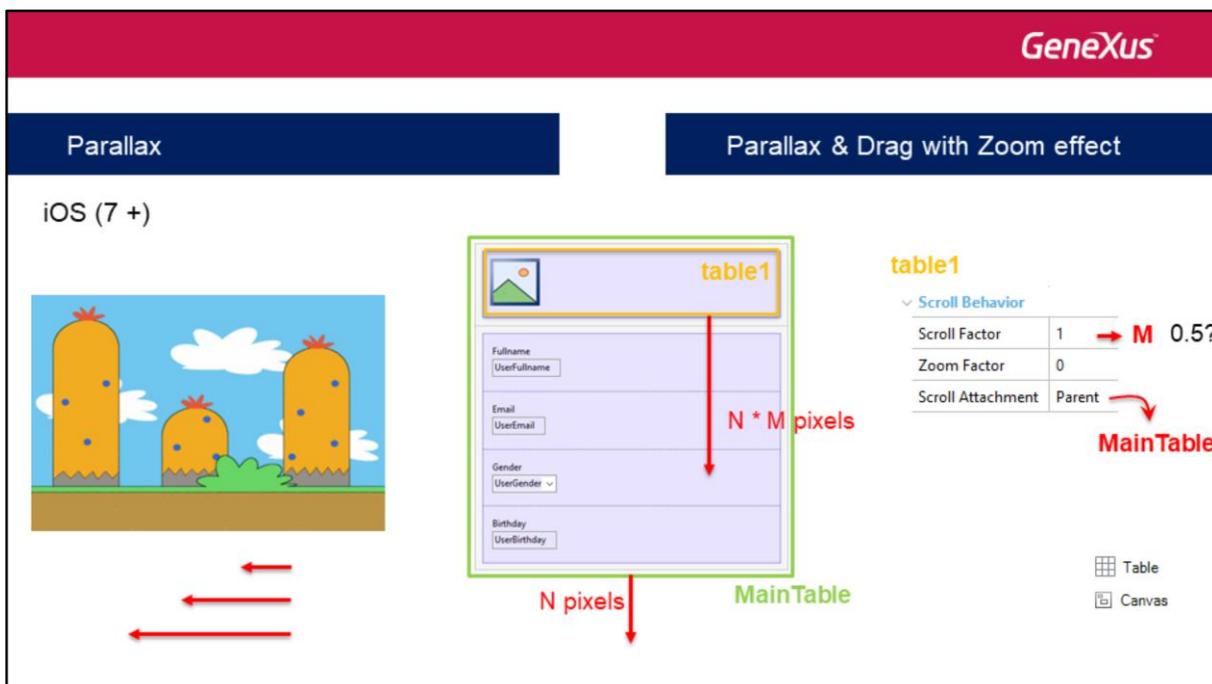


The screenshot displays a mobile application interface on the left and a properties panel on the right. The interface shows a calendar for August 2016 with a red highlight on the 24th. Below the calendar is a list of events: "Drop off dry cleaning" (8:00 AM - 9:30 AM), "Team Meeting" (11:00 AM - 12:00 PM), "Lunch with Tom" (12:00 PM - 1:00 PM), "Hair Appointment" (6:00 PM - 9:00 PM), and "Drinks with Friends!" (8:30 PM - 9:30 PM). The bottom navigation bar includes "Today", "Calendars", and "Inbox".

The properties panel on the right is titled "Properties" and shows the class "Application". The "Action Tint Color" property is highlighted with a red box and set to the hex value "#FF3B30".

Class: Application	
Name	Application
Description	
Background Color	
Background Image	(none)
Background Image Fill Mo	Scale To Fill
Placeholder Image	(none)
Image Loading Indicator	False
Date Picker Image	(none)
Prompt Image	(none)
Application Colors	
Primary Color	
Primary Color Dark	
Accent Color	
Control Normal Color	
Control Activated Color	
Control Highlight Color	
Action Tint Color	#FF3B30

Aquí vemos un ejemplo de esta propiedad en acción.



El efecto Parallax es un efecto visual utilizado frecuentemente en aplicaciones para Smart devices y también en páginas web. Consiste en desplazar un objeto por sobre otro que está por detrás, a distinta velocidad uno del otro, dando la sensación de profundidad.

En el ejemplo vemos un efecto de paralaje con tres capas que se mueven a distintas velocidades. La más interior es la que se mueve más lento.

En GeneXus el efecto de paralaje se consigue para una **tabla** o **canvas** y su contenido, en relación al contenedor u a otros controles especificados en la propiedad Scroll Attachment.

Por ejemplo, si tenemos una table1 con una imagen, y esa table1 está contenida en una MainTable, y queremos que cuando se hace scroll sobre esta MainTable, por ejemplo hacia abajo, por cada N pixels que esa MainTable y todo su contenido se desplacen hacia abajo, la table1 se desplaza $N * M$ pixels, entonces debemos editar las propiedades de la table1 y bajo el grupo Scroll Behavior modificar el valor por defecto 1 de la propiedad Scroll Factor por M.

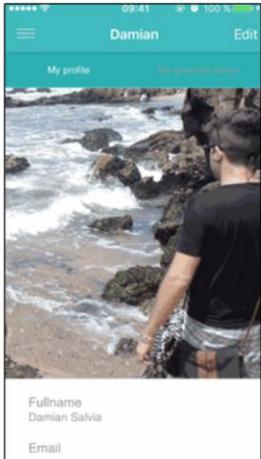
Ese factor de scroll lo es en relación al control o controles que se especifiquen en la propiedad Scroll Attachment. Por defecto es el contenedor de esta tabla, que en nuestro caso coincide con MainTable, pero pueden ser una lista de controles.

Observemos que si $M > 1$ entonces al hacer scroll sobre MainTable la table1 se desplazará en la misma dirección pero más rápido que el resto del contenido de MainTable. En cambio si $M < 1$, entonces será al revés.

Si $M = 0.5$ entonces table1 y su contenido se desplazarán a la mitad de velocidad que el resto de los controles de MainTable, por lo que si el scroll es hacia abajo, quedará un espacio vacío entre la imagen y los atributos que le siguen. Allí podemos utilizar la otra propiedad, Zoom Factor, para hacer que la table1 y su contenido hagan un zoom-in para ocupar más espacio en ese caso.

Parallax

iOS (7 +)



Parallax & Drag with Zoom effect

table



Scroll Behavior	
Scroll Factor	0.5
Zoom Factor	1
Scroll Attachment	MainTable

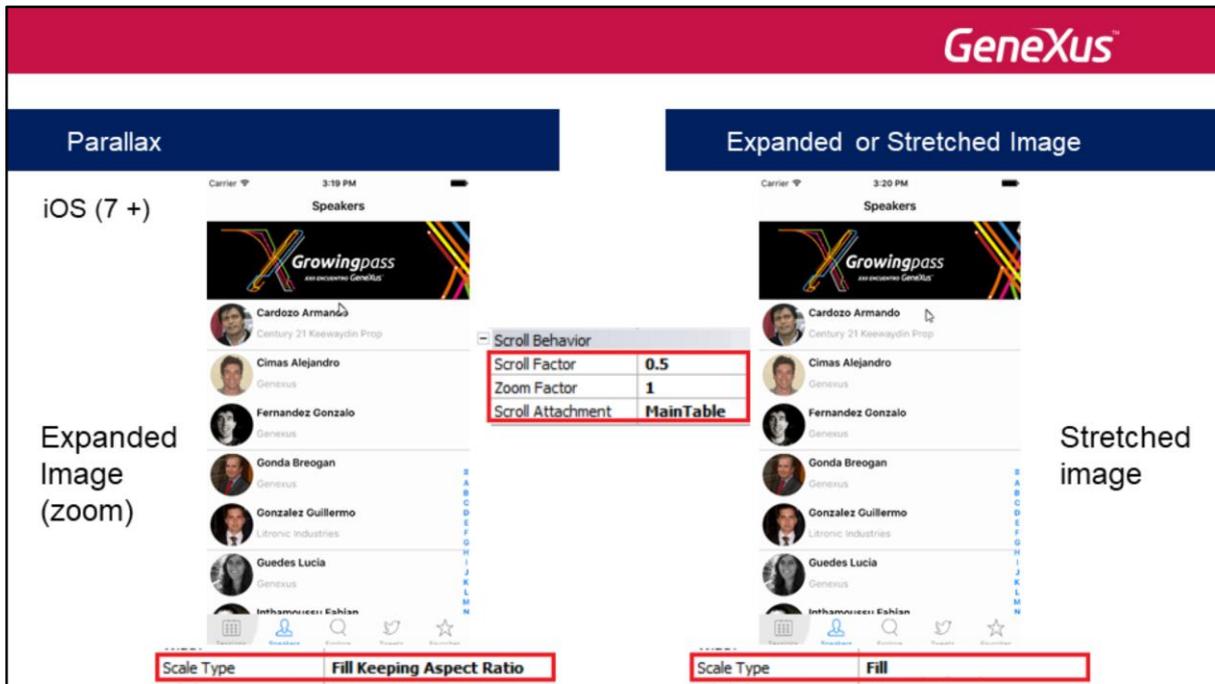
Table
Canvas

El efecto de Drag with Zoom consiste en expandir un control (por ejemplo una imagen) de fondo, cuando el usuario hace un Drag hacia abajo, al control que está sobre él. Como el control que está por arriba al desplazarse hacia abajo deja lugar, la imagen del fondo puede expandirse con un zoom, ocupando ese lugar.

Para conseguir estos efectos, usamos la propiedad Scroll Attachment y combinamos las propiedades Scroll Factor y Zoom Factor de las tablas y los canvases.

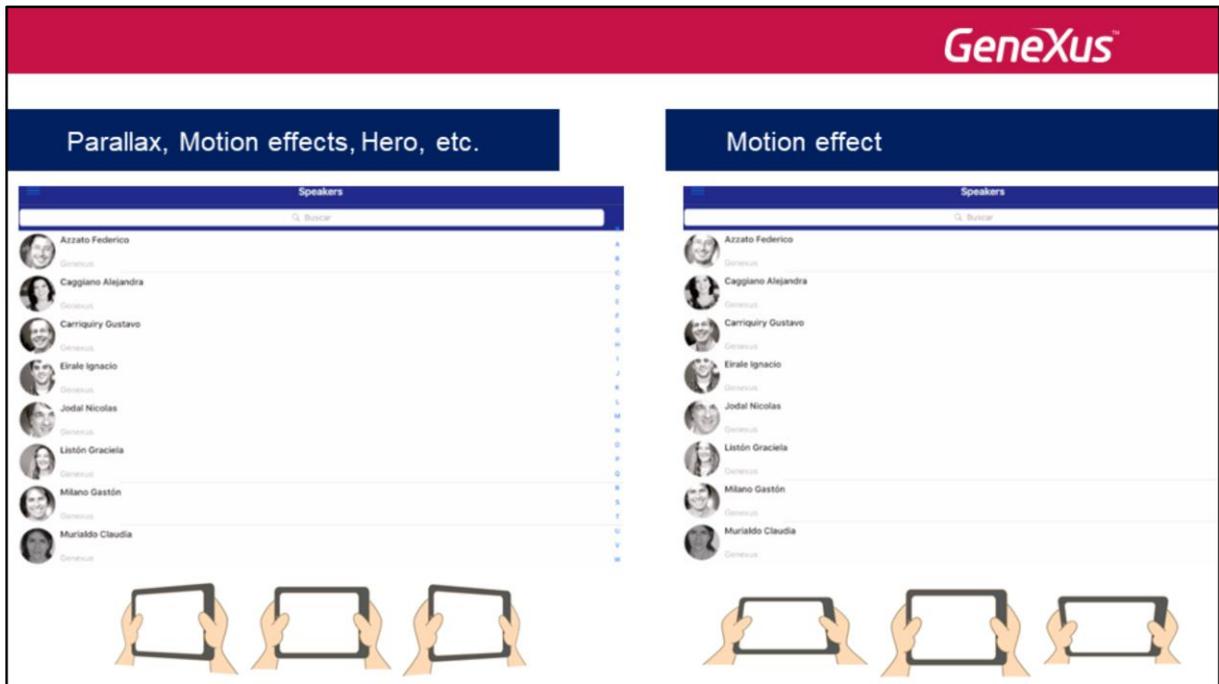
La propiedad Scroll Factor ya la vimos. El Zoom Factor es un valor decimal (por defecto 0, es decir, no zoom) que indica qué tanto zoom in/out hay que aplicar al control cuando el usuario hace scroll sobre los controles de la propiedad Scroll Attachment. Valores positivos o negativos provocarán que sea un zoom in o out.

Este efecto solamente es soportado en iOS 7 o superior.



Un caso particular es al hacerse un drag o pull down con una tabla o canvas que contiene una imagen. Podemos querer que en lugar de expandirse en todas direcciones (hacerse un zoom) como vimos antes, solamente se estire.

Esto se logra utilizando los mismos valores de las propiedades Scroll Factor y Zoom Factor que para hacer un zoom en la imagen, sólo que para que la imagen se estire en lugar de expandirse cambiamos el valor de su propiedad Scale Type al valor Fill.



De la misma manera en que Android da la sensación de profundidad con la propiedad elevation, iOS lo hace a través de capas y usando el giroscopio. Este efecto es llamado Motion Effect.

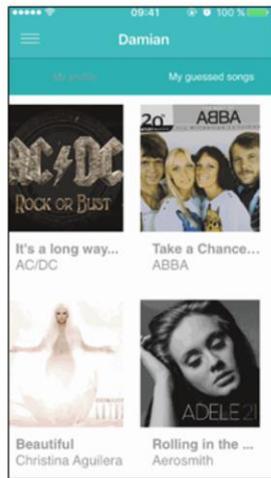
El efecto de movimiento se logra rotando el dispositivo sobre su eje horizontal o vertical, lo que provoca un movimiento en el control, haciendo que éste se mueva como si no estuviera fijo al dispositivo.

Si el movimiento es horizontal, la imagen se desplaza hacia la derecha o la izquierda, en sentido contrario al movimiento, dando la sensación de profundidad.

Si el movimiento es vertical, el desplazamiento es hacia arriba o hacia abajo. También es posible combinar ambos movimientos, cambiando la posición del dispositivo en ambas direcciones a la vez.

Parallax, Motion effects, Hero, etc.

iOS (7 +)



Motion effect

Group Table Canvas
Image Calendar

Motion Effect

Max Horizontal Offset 40
Max Vertical Offset 40



Para lograr el efecto de movimiento, se utilizan las propiedades Max Horizontal Offset y Max Vertical Offset de cualquiera de las clases listadas. Por defecto están en 0.

La propiedad Max Horizontal Offset indica el máximo desplazamiento que el control puede alcanzar cuando el usuario inclina el dispositivo horizontalmente hacia la derecha o hacia la izquierda. Se interpreta como un factor de movimiento, a mayor valor, mayor movimiento. Análogamente, la propiedad Max Vertical Offset indica el máximo desplazamiento del control, cuando el usuario inclina el dispositivo hacia adelante o hacia atrás.

Un valor negativo en estas propiedades, indica que el control se mueve en dirección opuesta al movimiento del dispositivo físico.

Un valor de cero implica que no hay efecto, y un valor positivo hará que el movimiento sea en la misma dirección que toma el dispositivo.

Live editing