

GeneXus[™]
The power of doing.

Theme & Images

User Interface & User Experience

GeneXus™ 16

- 1 Theme Object
- 2 Color Pallette
- 3 Platforms Setup
- 4 Images Setup

En este video comenzaremos estudiando el Objeto Theme y algunas de sus características principales , también veremos el uso de la paleta de colores. Luego continuaremos con la configuración de los temas en las plataformas Android e iOS y por ultimo veremos la configuración de las imágenes.

Theme & Images GeneXus™

Control Class

Control

Specific properties
(for that control only)

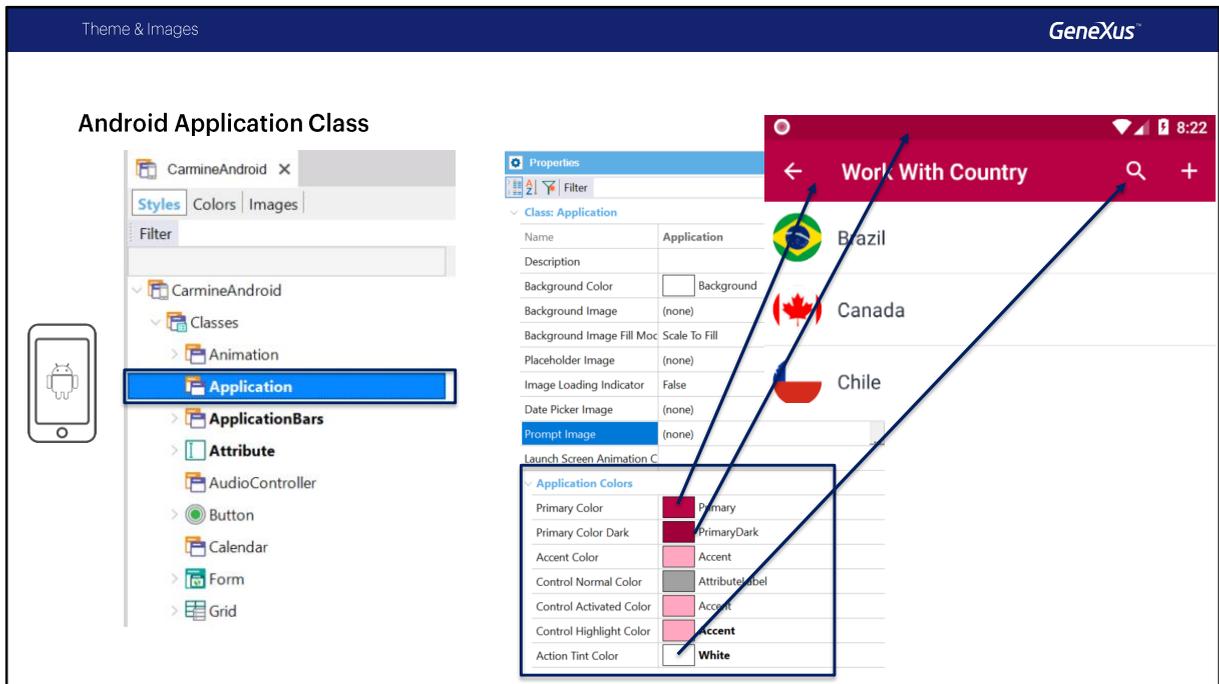
Class
Property

Theme

Como sabemos, en toda pantalla tenemos controles (atributos o variables) y otros controles como imágenes, botones, grids, etc. Cada control tiene sus propiedades personalizables, que indican, por ejemplo, propiedades de la etiqueta, visibilidad, si esta habilitado.

Pero además, cada control tiene una clase asociada. Es en la clase donde se especifican otras propiedades, como colores, la fuente a utilizar por ejemplo. Las clases permiten definir una sola vez algunos aspectos, para poder aplicárselos a muchos controles (del mismo tipo) a la vez.

Como la estética y el comportamiento general de la app debe ser uniforme y acorde a las guías de cada plataforma, es en el objeto Theme, que contiene todas las clases predefinidas y las que agreguemos, en el que haremos buena parte del trabajo de diseño y comportamiento para lograr la UI y UX que los diseñadores han determinado.



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.

Por defecto esas propiedades vienen pre configuradas y son las que vimos cuando ejecutamos la aplicación por primera vez, vemos que el Primary color se utiliza para el Application Bar, el color Primary Color Dark se utiliza en el Status Bar y por ejemplo las acciones utilizan el color Action Tint Color.

iOS Application Class

The screenshot displays the GeneXus IDE interface for configuring the iOS Application Class. On the left, the 'Classes' tree shows the 'Application' class selected. The 'Properties' window on the right lists various properties for the 'Application' class, including 'Name', 'Description', 'Background Color', 'Background Image', 'Background Image Fill Mode', 'Placeholder Image', 'Image Loading Indicator', 'Date Picker Image', and 'Prompt Image'. Under the 'Application Colors' section, the 'Action Tint Color' is set to a red color with the hex code #FF3B30. A blue arrow points from this property to a red square in the application preview on the right, which shows a calendar interface with a red date marker on the 24th of August.

Name	Application
Description	
Background Color	
Background Image	(none)
Background Image Fill Mode	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

Para iOS la única propiedad del grupo Application Colors que tiene efecto es Action Tint Color. Este será el color que asumirán la acciones.

Custom Classes

The screenshot displays the 'Custom Classes' section in the GeneXus Theme & Images editor. On the left, a list of classes is shown, including **TableTrackBullet** and its subclasses **TableTrackBullet001** through **TableTrackBullet010**. Below this list, a vertical list of other classes is visible, with **AttributeTrackTitle** highlighted in blue. On the right, the properties for the selected class are shown in a table format.

Background Color	TrackColor001
Highlighted Background Color	TrackColor001
Background Color	TrackColor003
Highlighted Background Color	TrackColor003
Margin Left	30dip
Margin Bottom	
Margin Right	30dip
Background Color	
Highlighted Background Color	
Background Image	(none)
Highlighted Background Image	(none)
Forecolor	White
Highlighted Forecolor	
Font	_ _ 14dip _ SanFrancisco _
Font Weight	
Font Style	
Font Size	14dip
TextDecoration	
Font Family	San Francisco
Font Category	

Como desarrolladores, a partir de la información que nos enviará el diseñador gráfico, modificaremos o crearemos clases y subclases en los themes, que luego serán utilizadas por los controles que insertemos en los layouts.

Aquí estamos viendo un ejemplo que luego reflejaremos en nuestra app: la clase **TableTrackBullet** y sus subclases, cada una con un color característico de cada Track, o la clase **AttributeTrackTitle** para mostrar los nombres de los Tracks.

Theme & Images GeneXus™

Elevation Property

Valid Controls:
 Attribute/Variable, Button, Grid,
 Group, Image, Tab, Table,
 Text Block

Value	by default	5	10	20
Effect				



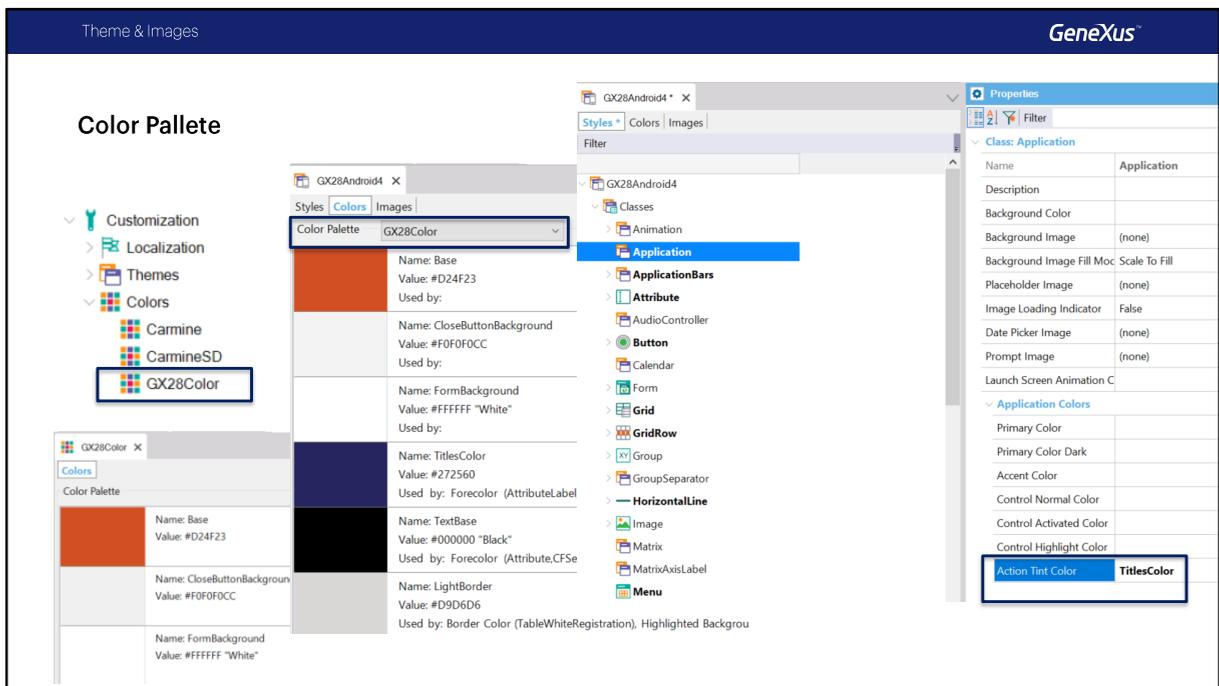
- TableUpcoming**
- TableUserData
- TableVideo
- TableVideosMissed
- > TableWhite
- TableWidgetTitle
- < Table...

Background Image Fill Mode	Scale To Fill
Elevation	3
Padding	4dip
Padding Top	4dip
Padding Left	4dip
Padding Bottom	4dip
Padding Right	4dip

O, por ejemplo, podemos especificar que controles de la app Android (de algunos de los tipos que se indican), puedan tener un sombreado, de acuerdo al valor de la propiedad Elevation. De este modo se estarán siguiendo las guías de Material Design.

Aquí vemos como se vería una imagen con un valor por default de la propiedad Elevation (o sea sin especificar), con un valor de 5, de 10 y de 20.

Particularmente en la app Event Day lo utilizaremos por ejemplo en la clase TableUpcoming, para mostrar ciertos contenidos como la Lista de Oradores Destacados que vemos a la derecha.



Los diseñadores suelen definir una paleta de colores, y luego aplicar esa paleta al diseño general. Podemos crear un objeto de tipo paleta, definir en él los colores, dándoles un nombre, y luego al Theme asociarle en su pestaña Colors esa paleta, para poder utilizarla luego en las clases.

Esto no es obligatorio, pero es una buena práctica, para no tener que estar ingresando siempre los colores en las clases por su valor, sino hacerlo directamente por su nombre, además podríamos crear varias paletas y redefinir los colores en cada una haciendo independiente al Theme de cada color en particular.

Theme & Images GeneXus™

Multiplatform

Platform: Any Android

Name	Any Android
OS	Android
Version	
Device Kind	All
Size	All
Theme	CarmineAndroid
Additional Themes	(none)
Navigation Style	Default
Default Layout Orientation	Any
Bounds Name	
Minimum Shortest Bound	0
Maximum Shortest Bound	0

Platform: Any iOS

Name	Any iOS
OS	iOS
Version	
Device Kind	All
Size	All
Theme	CarminelOS
Additional Themes	(none)
Navigation Style	Default
Default Layout Orientation	Any
Bounds Name	
Minimum Shortest Bound	0
Maximum Shortest Bound	0
Minimum Longest Bound	0
Maximum Longest Bound	0

Customization

- Localization
- Themes
 - Carmine
 - CarmineSD
 - CarminelOS
 - CarmineAndroid
 - SD Platforms
 - Any Platform
 - Any Android
 - Android Phone
 - Android Tablet 7"
 - Android Tablet 10"
 - Any iOS
 - iPad
 - iPhone
 - iPhone 3.5"
 - iPhone 4"
 - iPhone 4.7"

Decíamos que nuestras apps para Smart Devices serán multiplataforma. Es decir, creamos básicamente la misma app y haremos el mínimo trabajo indispensable para personalizarla de modo que se ejecute en dispositivos Android y en dispositivos iOS, siguiendo sus guías de diseño y comportamiento.

Contaremos en cada Base de Conocimiento con dos themes ya creados, CarmineIOS y CarmineAndroid, y por default se asocian estos Themes a la plataforma Android, y otro para la plataforma iOS, y el será el mismo para toda la plataforma independientemente de si se trata de Teléfono o Tablet, y de su tamaño y resolución, pero podríamos crear otros themes para hacer variar el diseño y comportamiento general dependiendo de esas dimensiones.

Demo: Configuration of Themes and Platforms

Vamos a ver todo esto en GeneXus.

Primero vamos a ver que ya tenemos en la KB los temas creados, GX28Android4 y GX28IOS, también tenemos la paleta de colores GX28Color.

Vamos a abrir el tema GX28Android4, vamos a ver en la solapa Colors, en esta propiedad Color Pallette podemos cambiar y asignar la paleta de colores a un tema específico.

Como recomendación, usar nombres que no hagan referencia al color, por ejemplo FormBackground.

Bien, vamos ahora a en la clase Application el Action Tint Color y vamos a hacer referencia a un nombre de la paleta, TitlesColor, también podemos ingresar y seleccionar un color de la paleta. Bien vamos a grabar.

Vamos a ir ahora a la plataforma Any Android y vamos a asociar el tema, seleccionamos Gx28Android4, en este caso para todas las plataformas Android va a ser el mismo. Lo mismo hacemos en Any IOS, seleccionamos GX28 IOS.

Bien. Antes de ejecutar vamos a cambiar en el objeto EventDay, en el menú, Menu for Smart Devices, la propiedad Android Base Style, vamos a a usar Light en vez de Light with Dark Action Bar. Esto nos va a dar el efecto que estamos buscando en la aplicación.

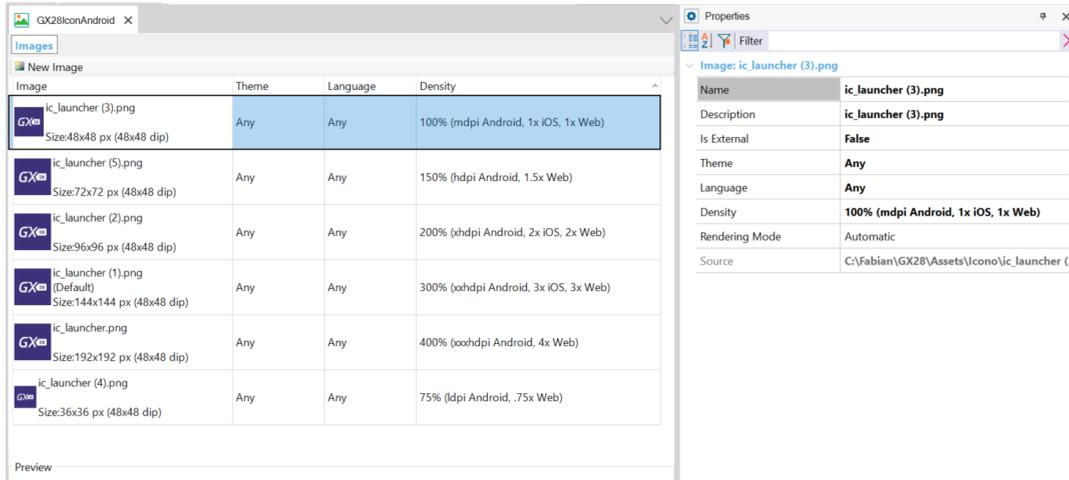
Vamos a correr la aplicación, bien, esperamos un momentito.

Bien. Ya tenemos la aplicación instalada en el emulador, vemos que cambio el color Rojo, ese tono de Rojo en el Application Bar, vamos a ver por ejemplo, tenemos las acciones que están tomando el Action Tint Color.

Vamos a ver por ejemplo Uruguay, aca vemos por ejemplo la propiedad Tabs Strip, en esa tonalidad verde, no tiene un color especifico porque esta tomando el valor default de la plataforma.

Con esto termina la demo, vamos a seguir viendo los siguientes temas.

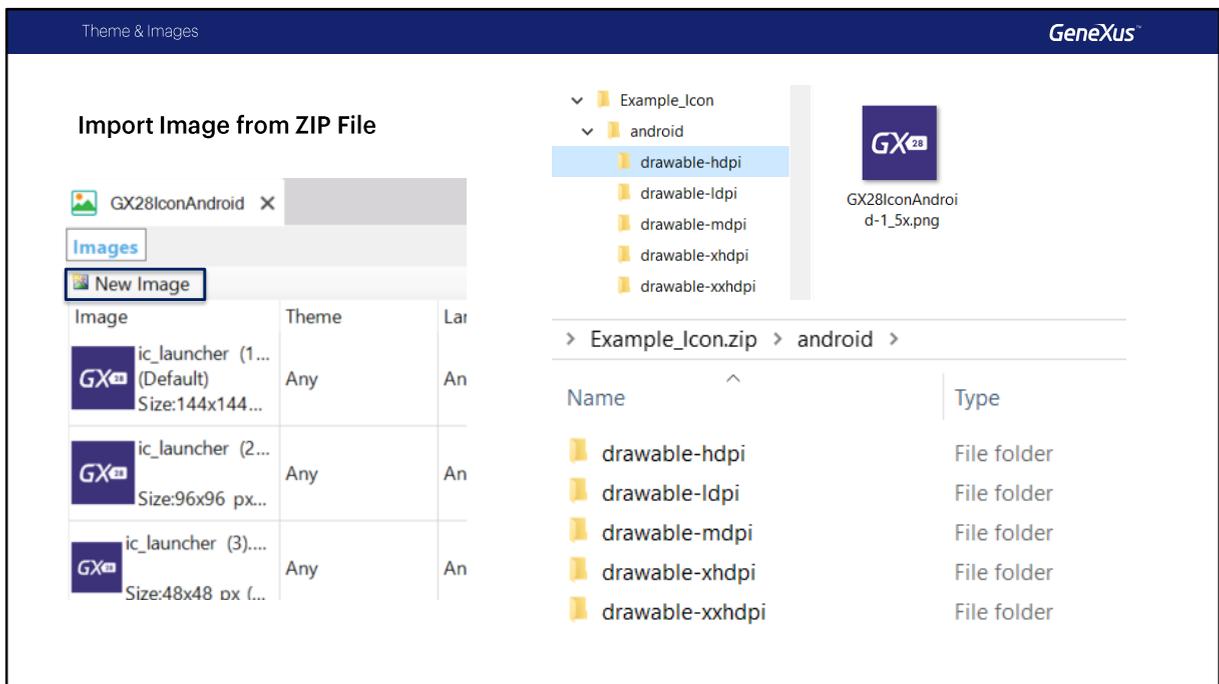
Setting up an Image



Vamos a ver algunos aspectos sobre las imágenes.

Aquí tenemos el ejemplo del ícono de la aplicación. Observemos que hemos creado un único objeto de tipo imagen en la KB, al que hemos llamado GX28IconAndroid y dentro hemos ingresado cada archivo con la imagen que nos mandó el diseñador, pudiendo variar las mismas por Theme, densidad y lenguaje.

En este caso, en el objeto main de la app, simplemente configuramos este objeto imagen y dependiendo de la plataforma y de la resolución del dispositivo, se utilizará la imagen que corresponda.



Para agregar las distintas versiones de una misma imagen tenemos dos opciones.

La primera consiste en crear una nueva imagen y luego agregar manualmente con la opción New Image las demás versiones y configurar para cada una las densidades correspondientes.

La otra es, Importar las imágenes desde un archivo ZIP, en este caso las imágenes deben estar organizadas dentro del archivo ZIP como se muestra en la imagen Example_Icon.ZIP, debe tener un folder Android y luego para cada densidad un folder con la imagen correspondiente.

Luego Creamos una nueva imagen utilizando la opción Create Image from File, seleccionamos el archivo ZIP, en nuestro caso Example_Icon.ZIP y presionamos el botón Finalizar.

El resultado es el que vemos en esta imagen de la derecha, donde ya todas las densidades quedan configuradas automáticamente al importar el archivo ZIP. Tengan en cuenta esta característica al solicitar las imágenes al diseñador, quien ya se las puede proveer organizadas de una manera que puedan importarlas fácilmente a la base de conocimiento ahorrando mucho tiempo.

Setup Main Object Images



▼ **Android**

Android Version Code	1.0
Android Version Name	1.0
Update URL	
Android Package Name	com.artech.eventday.eventday
Android Application Icon	GX28IconAndroid
Android Notification Icon	(none)
Android Portrait Launch Image	GX28LaunchAndroid
Android Landscape Launch Image	GX28LaunchAndroidLandscape



▼ **Apple**

Apple Version Code	1.0
Apple Version Name	1.0
App Store URL	
Apple Bundle Identifier	com.artech.EventDay
Development Team ID	
Devices	iPhone & iPad
Watch Main	(none)
Apple Application Icon	GX28Icon
Apple Launch Image	GX28Launch

Vamos a repasar esto que ya vimos.

Recuerden que es necesario disponer de las imágenes de **launch** (para la orientación Portrait y Landscape) y **el ícono de la aplicación**, que se configuran en las propiedades del objeto main (posiblemente un menú, como en el caso de nuestra aplicación) como vemos aquí.

En el caso de Android mínimamente necesitamos Android Application Icon, Android Portrait Launch Image y Android Landscape Launch Image, en el caso de Apple simplemente basta con ingresar Apple Application Icon y Apple Launch Image.

Bien, estas fueron algunas de las características mas importantes sobre el Objeto Theme, el tratamiento de imágenes y la configuración de las plataformas, durante el resto del curso iremos presentando algunas clases y propiedades específicas del desarrollo para Smart Devices, igualmente pueden profundizar sobre estos temas en la wiki de GeneXus.

GeneXus™

Videos	training.genexus.com
Documentation	wiki.genexus.com
Certifications	training.genexus.com/certifications