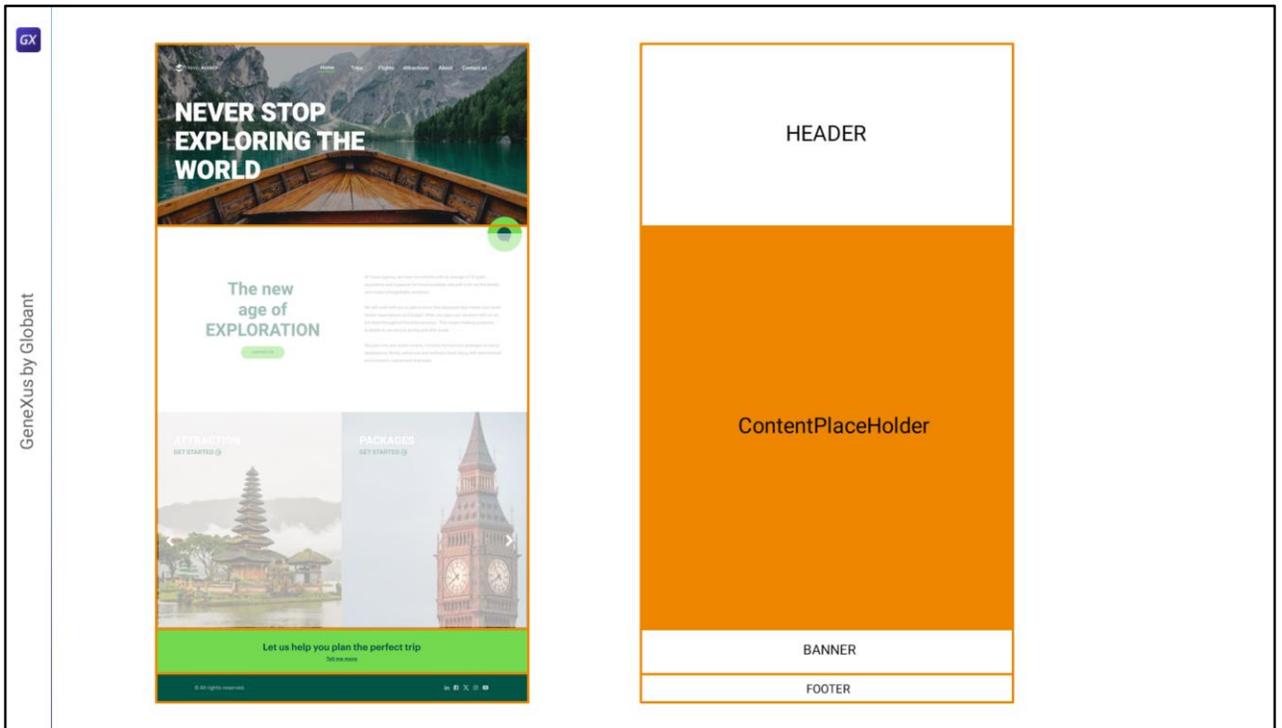


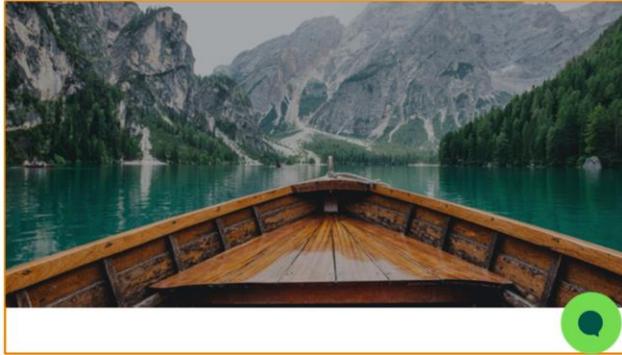
Canvas Control for Header



Cecilia Fernández



El objetivo que ahora perseguiremos será implementar el Header del Master Panel.



The new age of EXPLORATION

[CONTACT US](#)

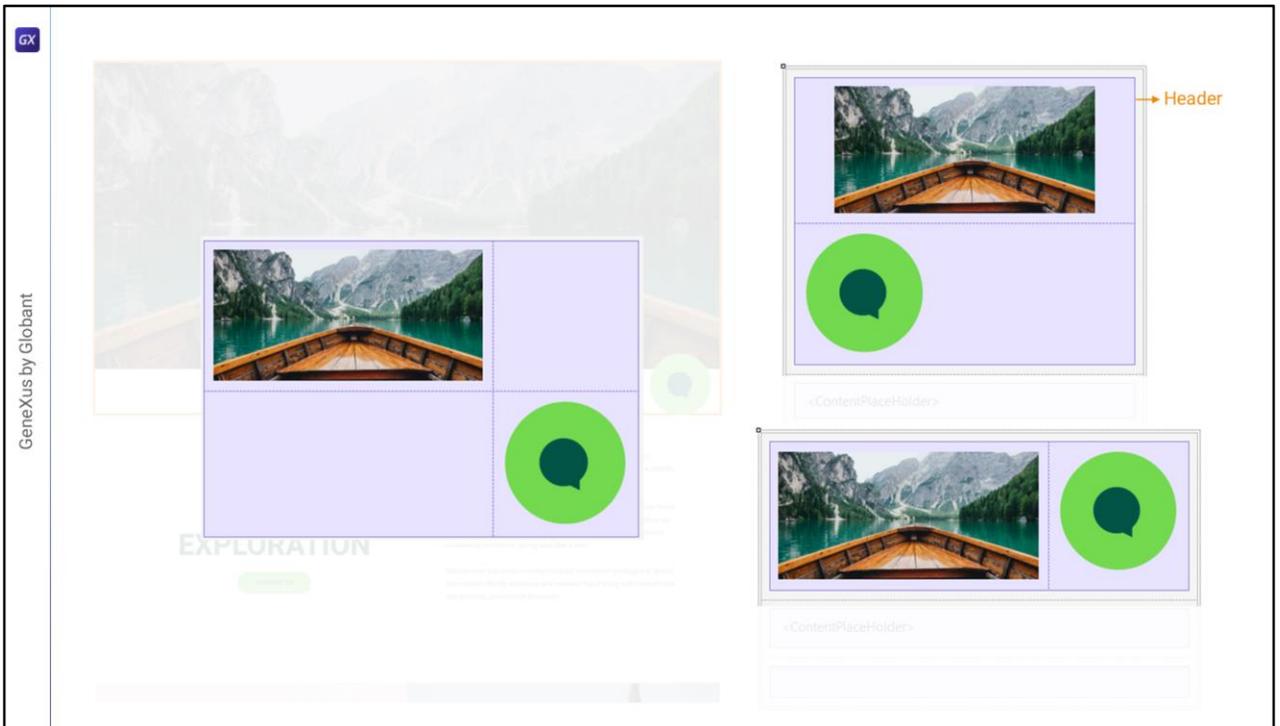
At Travel Agency, we have consultants with an average of 20 years experience and a passion for travel available who will work out the details and create unforgettable vacations.

We will work with you to plan a worry free adventure that meets your travel needs, expectations and budget. When you plan your vacation with us we are there throughout the entire process. This means making ourselves available to you before, during and after travel.

We plan river and ocean cruises, romantic honeymoon packages to sunny destinations, family, adventure and wellness travel along with international and domestic customized itineraries.

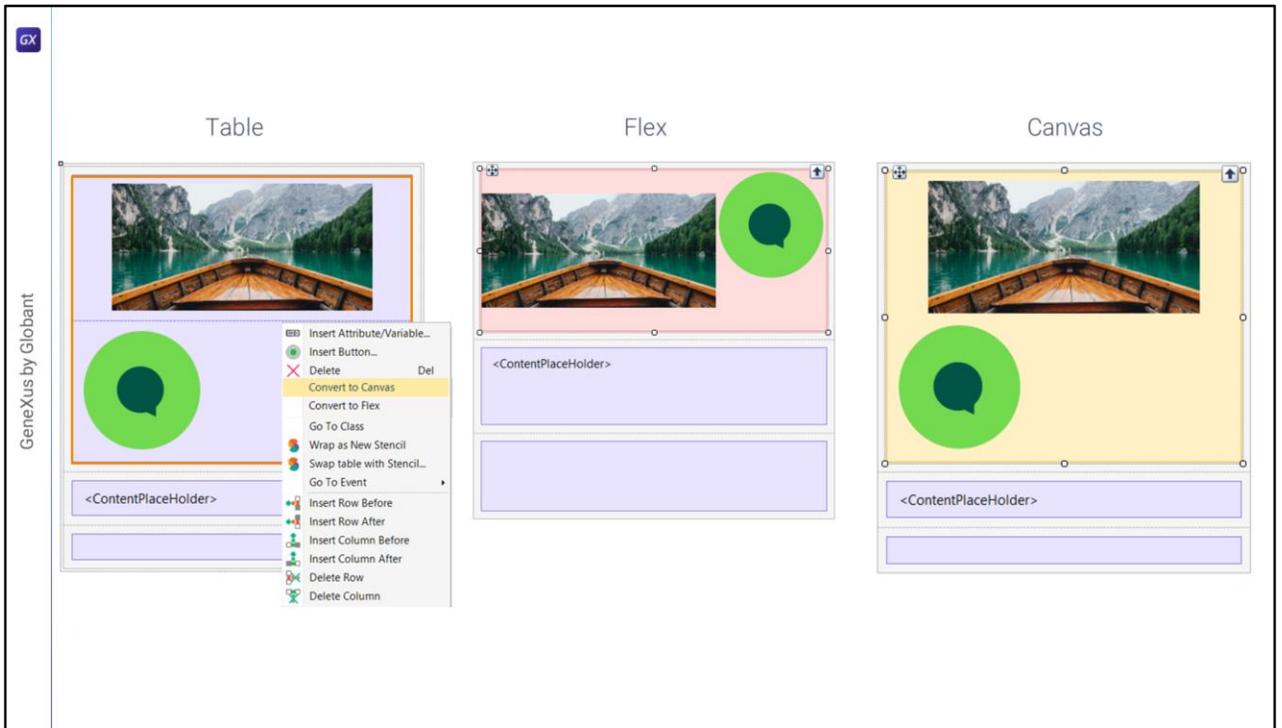
Los layouts, por default, contienen una única capa, en la que los controles ocupan cada uno su espacio, sin solaparse los demás. Cada control tiene su espacio reservado.

Así, por ejemplo, si empezamos en el Header por modelar únicamente la imagen de fondo y el botón del chatbot, para que puedan superponerse tendremos que ubicarlos a cada uno en una capa distinta.

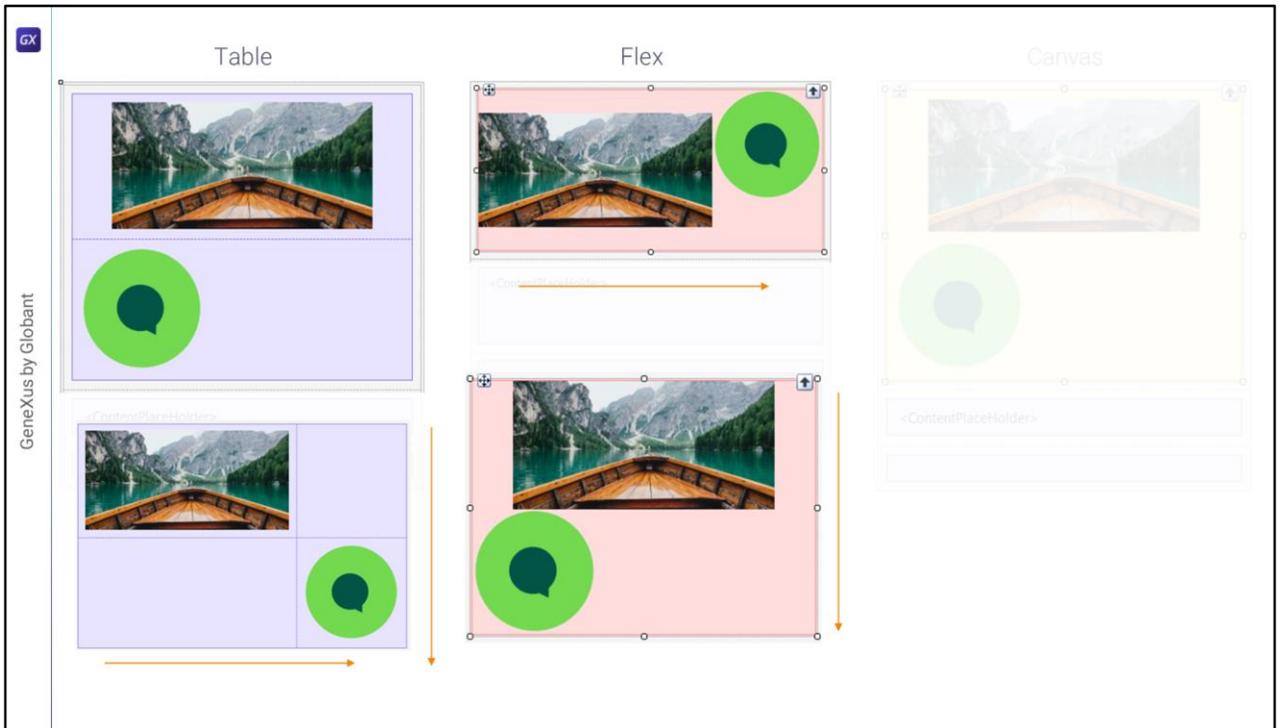


Teniendo en la fila 1 de la tabla Main una tabla (para darle el rol "Header" para la accesibilidad, como vimos en el video anterior), si arrastramos y soltamos allí dentro un control imagen, por ejemplo, para la imagen de fondo y un control botón para el chatbot (con esta imagen), necesariamente quedarán o en dos filas consecutivas o en dos columnas consecutivas, es decir, cada control ocupará un espacio disjunto en relación al otro (porque esas filas o columnas no se superponen).

Así, un control sólo podrá empezar donde el otro termina, ya sea vertical, ya sea horizontalmente, ya sea horizontal y verticalmente (para el caso de una tabla de 2x2 donde los controles se ubiquen en diagonal).

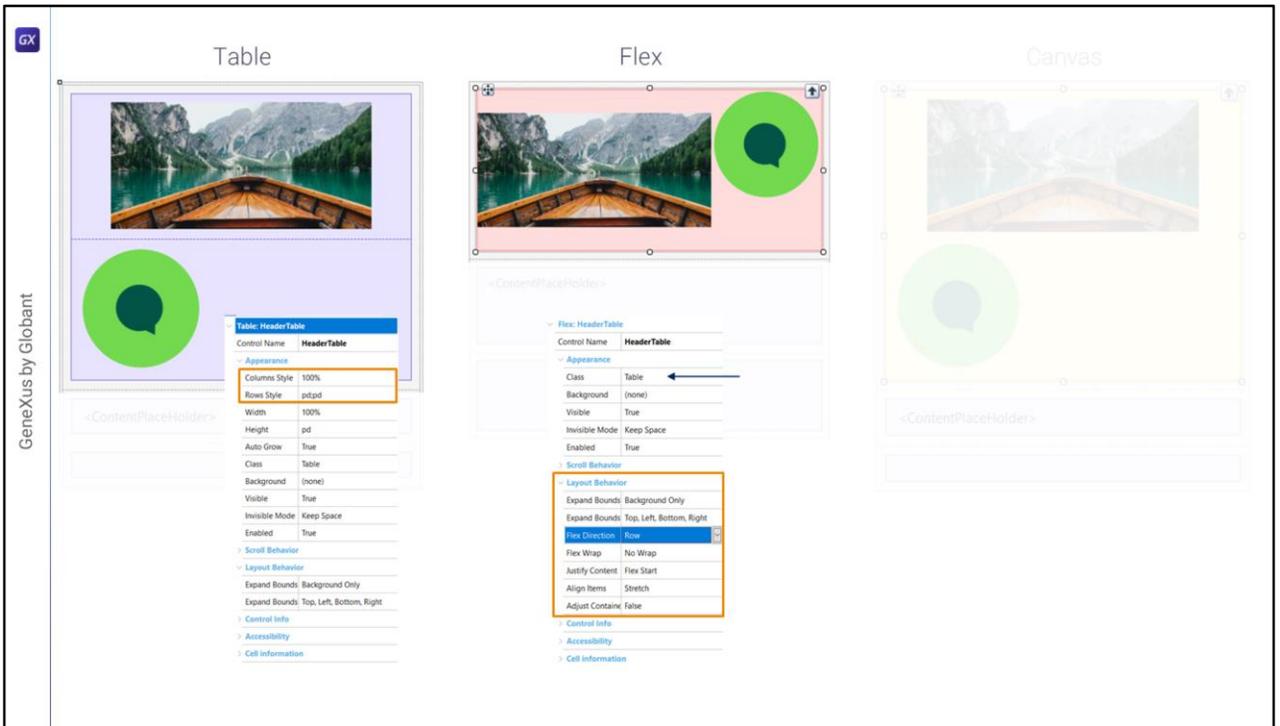


Podríamos transformar la tabla en contenedor Flex... o en contendor Canvas.

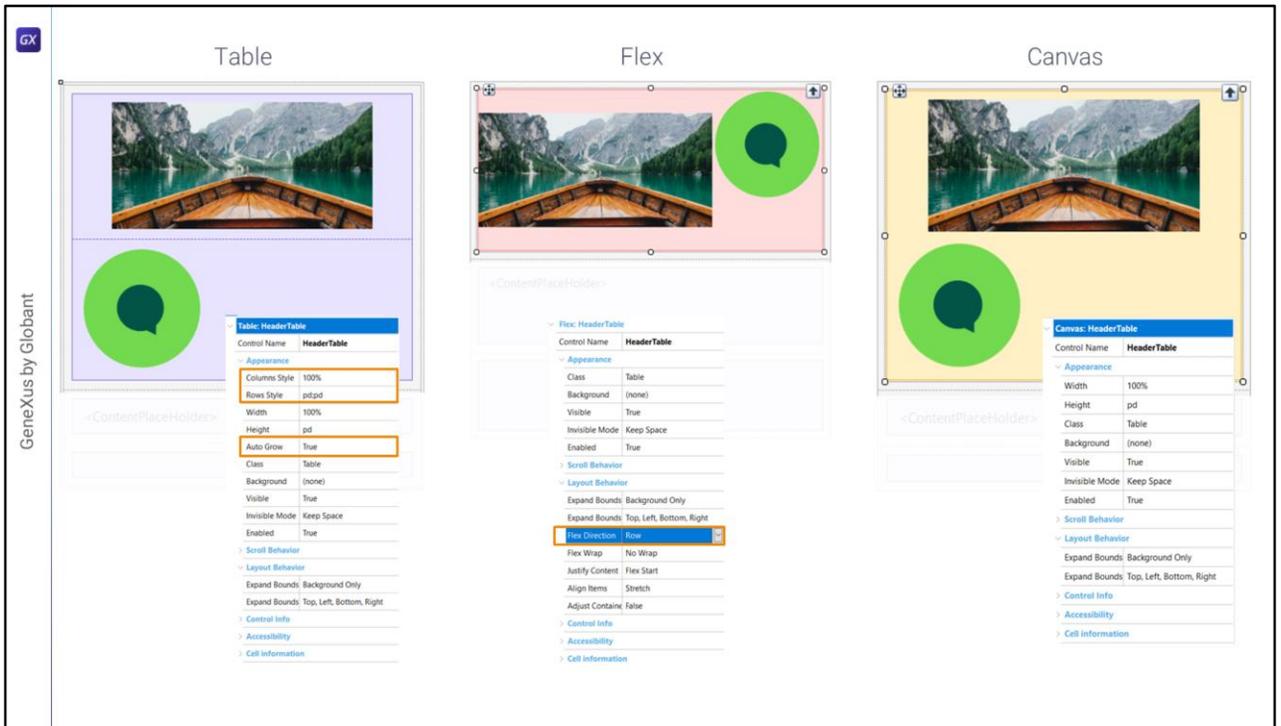


Si lo transformamos en contenedor Flex... la tabla, que es un control de dos direcciones (puede tener muchas filas y muchas columnas) se convierte en una especie de tabla flexible de una sola dirección:

- o una fila (dirección Row) en la que se suceden n controles, ya sea de izquierda a derecha o de derecha a izquierda
- o una columna (dirección Column) en la que se suceden n controles.



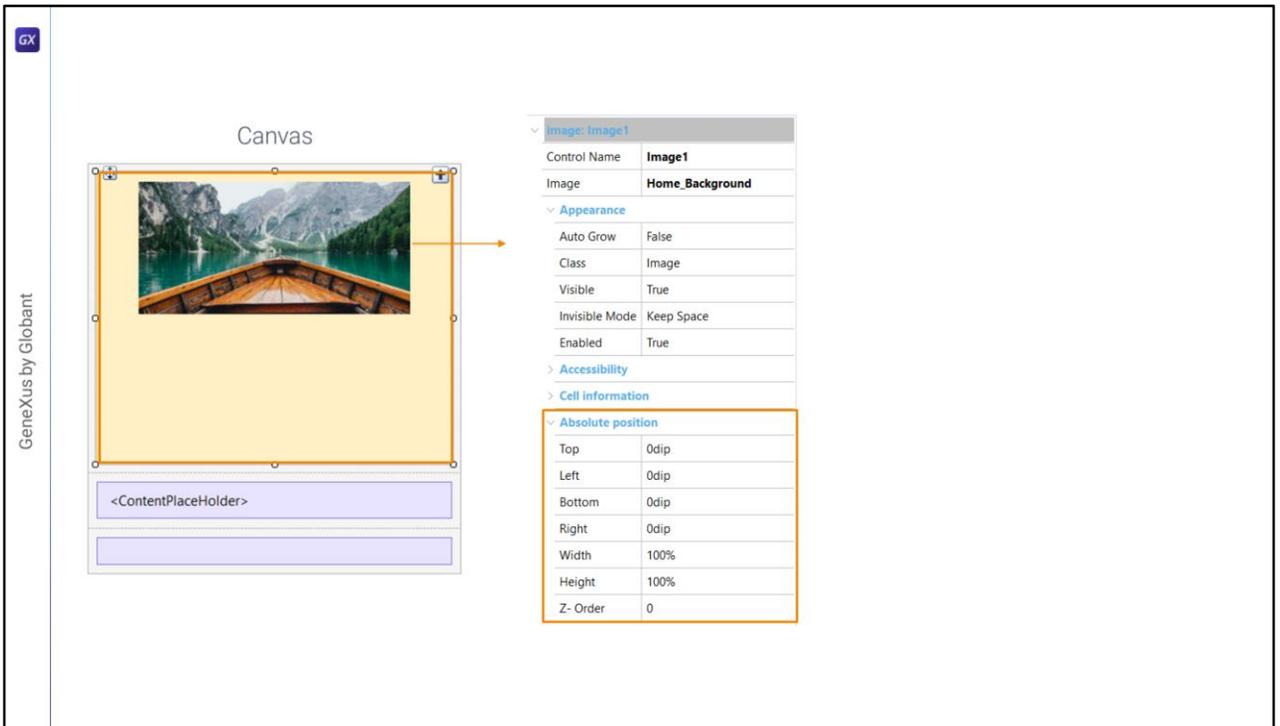
Así, vemos que desaparecen las propiedades Columns Style y Rows Style propias del control Table, y aparecen, en cambio, estas de aquí. En particular vemos cómo la propiedad Flex Direction establece si será un contenedor que ubica sus controles a lo largo de una fila o de una columna. Los controles que se ubiquen allí dentro se colocarán uno a continuación del otro, y el espaciado entre ellos tendremos que darlo a través de propiedades, como por ejemplo la gap, a nivel de la clase que le asociemos.



Pero para lograr que los controles internos al contenedor se puedan superponer necesitamos el otro tipo de contenedor, el Canvas.

Si comparamos las propiedades respecto al control tabla, vemos que tiene 3 propiedades menos: Columns Style y Rows Style desaparecen y no aparece ningún relevo (observemos que en el Flex sí, estaba la Flex Direction). Aquí no hay ninguna porque el concepto de fila y columna desaparece completamente. Es que será un contenedor donde todo lo que se coloque dentro estará posicionado de manera **absoluta** en relación a los **bordes de contenedor**, y, es independiente de la existencia de otros controles que también estén allí dentro. El concepto de filas y columnas pierde sentido.

También desaparece la propiedad Auto Grow, que para el Canvas será siempre True.



Lo que va a pasar, entonces, es que todos los controles que estén dentro del Canvas agregarán propiedades para el posicionamiento absoluto. Cada uno se va a posicionar en forma absoluta en relación a los bordes del Canvas.

Y eso quiere decir posicionamiento absoluto, que no depende en nada de los demás controles que estén allí dentro (ni afuera, por supuesto), sólo importa el Canvas como referente.

Por eso vean que posicionados sobre el control imagen aparece este grupo **Absolute position...**

GeneXus by Globant

Canvas

<ContentPlaceholder>

z

1

0

x

y

Image: Image1

Control Name	Image1
Image	Home_Background
Appearance	
Auto Grow	False
Class	Image
Visible	True
Invisible Mode	Keep Space
Enabled	True
Accessibility	
Cell information	
Absolute position	
Top	0dip
Left	0dip
Bottom	0dip
Right	0dip
Width	100%
Height	100%
Z- Order	0

Button: chatbot

Control Name	chatbot
On Click Event	'chatbot'
Caption	
Appearance	
Class	Button
Visible	True
Invisible Mode	Keep Space
Enabled	True
Format	Text
Image	Chatbot
Disabled Image	(none)
Image Position	Above Text
Control Info	
Accessibility	
Cell information	
Absolute position	
Top	0dip
Left	0dip
Bottom	0dip
Right	0dip
Width	100%
Height	100%
Z- Order	0

Y lo mismo para el control botón.

La última propiedad es la más sencilla de entender: es la que determina la capa en la que se encontrará el control.

Es que para lograr la superposición aparece la otra dimensión, la z, la que introduce capas, justamente. La capa más profunda será la de valor 0, y de allí en más podemos numerar las capas.

GeneXus by Globant

Canvas

Image: Image1	
Control Name	Image1
Image	Home_Background
Appearance	
Auto Grow	False
Class	Image
Visible	True
Invisible Mode	Keep Space
Enabled	True
Accessibility	
Cell information	
Absolute position	
Top	0dip
Left	0dip
Bottom	0dip
Right	0dip
Width	100%
Height	100%
Z- Order	0

Button: chatbot	
Control Name	chatbot
On Click Event	'chatbot'
Caption	
Appearance	
Class	Button
Visible	True
Invisible Mode	Keep Space
Enabled	True
Format	Text
Image	Chatbot
Disabled Image	(none)
Image Position	Above Text
Control Info	
Accessibility	
Cell information	
Absolute position	
Top	0dip
Left	0dip
Bottom	0dip
Right	0dip
Width	100%
Height	100%
Z- Order	1

En nuestro caso dejaríamos a la imagen con la capa 0 y colocaríamos el botón en la 1 para que esté encima.

Ahora atendamos al resto de las propiedades que están relacionadas de acuerdo a dos ejes: el x y el y.

Como dijimos, en lo que hace al posicionamiento de cada control dentro del Canvas no importan para nada los controles hermanos. Sólo importa el propio control y los límites del Canvas.

GeneXus by Globant

Canvas

The image shows the GeneXus IDE interface. On the left, a canvas labeled 'Canvas' contains a green chatbot button. Below the canvas is a coordinate system with 'x' and 'y' axes. In the center, the 'Image: Image1' properties panel is visible. On the right, the 'Button: chatbot' properties panel is shown, with the 'Absolute position' section highlighted by an orange border.

Image: Image1	
Control Name	Image1
Image	Home_Background
Appearance	
Auto Grow	False
Class	Image
Visible	True
Invisible Mode	Keep Space
Enabled	True
Accessibility	
Cell information	
Absolute position	
Top	0dip
Left	0dip
Bottom	0dip
Right	0dip
Width	100%
Height	100%
Z- Order	0

Button: chatbot	
Control Name	chatbot
On Click Event	'chatbot'
Caption	
Appearance	
Class	Button
Visible	True
Invisible Mode	Keep Space
Enabled	True
Format	Text
Image	Chatbot
Disabled Image	(none)
Image Position	Above Text
Control Info	
Accessibility	
Cell information	
Absolute position	
Top	0dip
Left	0dip
Bottom	0dip
Right	0dip
Width	100%
Height	100%
Z- Order	1

Así que analicemos el botón del Chatbot.

Estas son las propiedades relativas al posicionamiento del botón respecto al eje x. Y estas respecto al eje y.

Las de cada eje están relacionadas entre sí, por lo que cuando fijemos 2 de las propiedades, la 3era quedará automáticamente determinada también, en porcentaje.

GeneXus by Globant

Canvas

Absolute position	
Top	0dip
Left	0dip
Bottom	0dip
Right	0dip
Width	100%
Height	100%
Z- Order	1

<ContentPlaceholder>

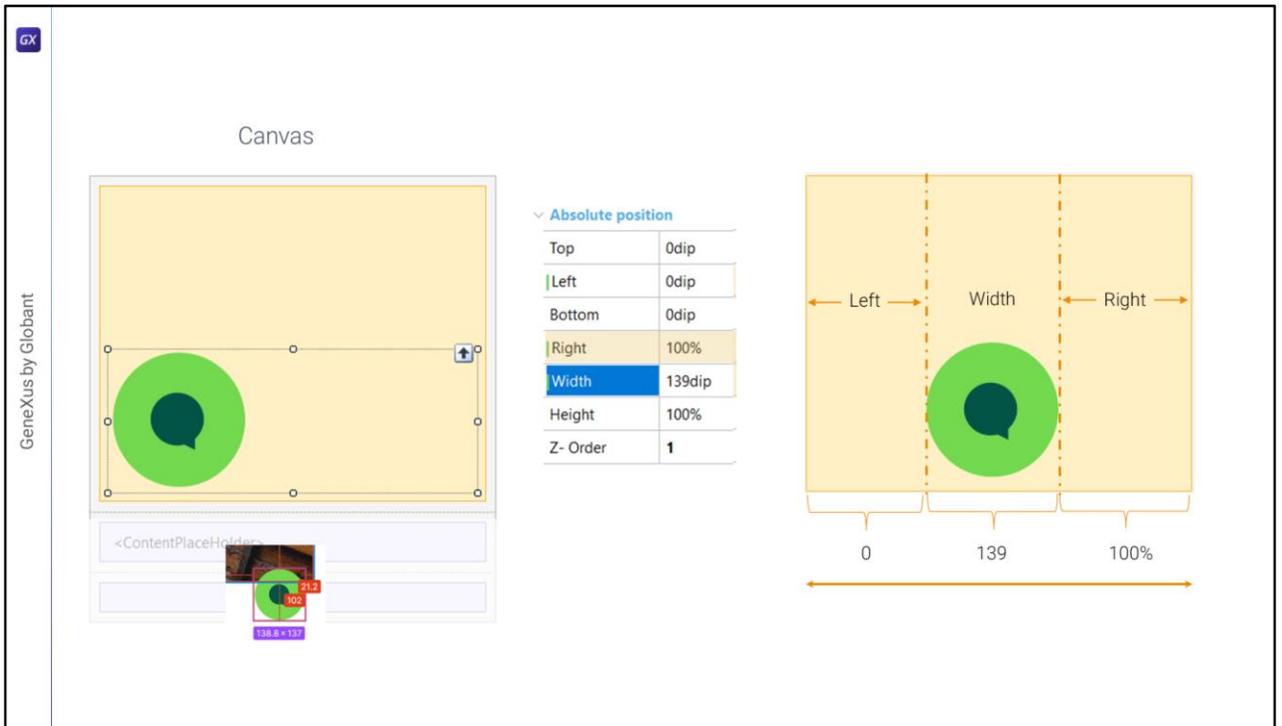
100%

Veámoslo con las del eje x.

Así estarán inicialmente, indicando: 0 dips respecto al borde izquierdo del Canvas, 0 dips respecto al borde derecho, y por tanto la del ancho del control quedará con 100% del ancho restante del Canvas, que en este caso corresponderá al ancho total.

En definitiva es como si se colocaran dos columnas a cada lado del control, y a la de la izquierda se le diera como ancho el del valor **Left**, a la de la derecha el valor **Right** y a la del medio, donde va el control, el valor **Width**.

Es evidente que fijando 2 la tercera queda determinada, porque sumadas deben dar 100% del ancho del canvas.



Si buscamos en Figma el ancho del botón y la distancia que lo separa del extremo derecho vemos esto.

Así, si fijamos el Width para que sea de 139 dips (o píxeles) al salir del campo automáticamente veremos este cambio en la propiedad Right. ¿Qué indican los valores actuales? Que el control botón estará a 0 dips del borde izquierdo del Canvas, es decir, pegado, se extenderá con un ancho de 139 dips y lo que lo distanciará del borde derecho del canvas será el 100% de tamaño restante, en relación al ancho del canvas.

GeneXus by Globant

Canvas

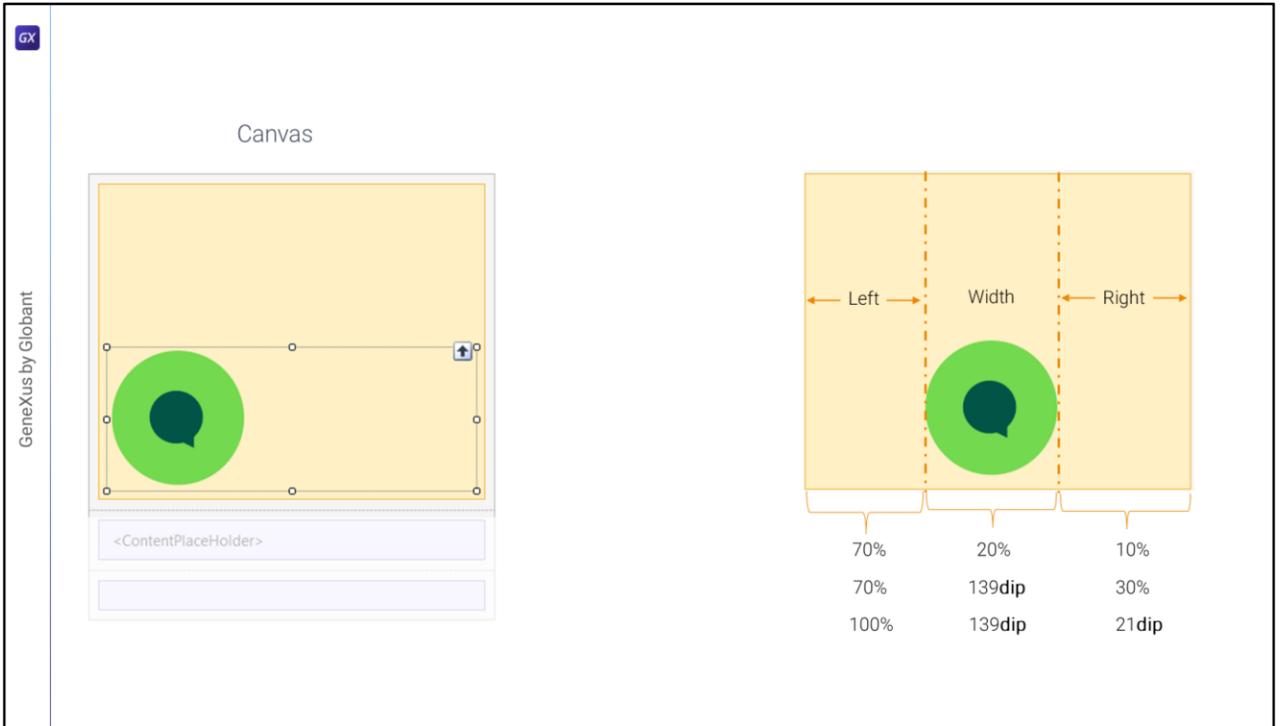
Absolute position	
Top	0dip
Left	100%
Bottom	0dip
Right	21dip
Width	139dip
Height	100%
Z-Order	1

Left Width Right

100% 139 21

Sin embargo nosotros queremos que esté a 21.2 dips (redondemos a 21) del borde derecho del Canvas. Por tanto al especificar esa distancia de la derecha, automáticamente cambia a 100% la distancia respecto al borde izquierdo.

Otra vez: será el 100% de lo que resulte de restar del ancho del Canvas los valores fijos de 139 dips y 21 dips.



Left, Width y Right pueden estar los 3 dados en porcentajes (y deberán sumar 100%), pueden estar dos dados en porcentajes (y deben sumar 100 %), o uno sólo, en cuyo caso siempre será 100%.

GeneXus by Globant

Canvas

Absolute position	
Top	0dip
Left	100%
Bottom	0dip
Right	21dip
Width	139dip
Height	100%
Z-Order	1

Top

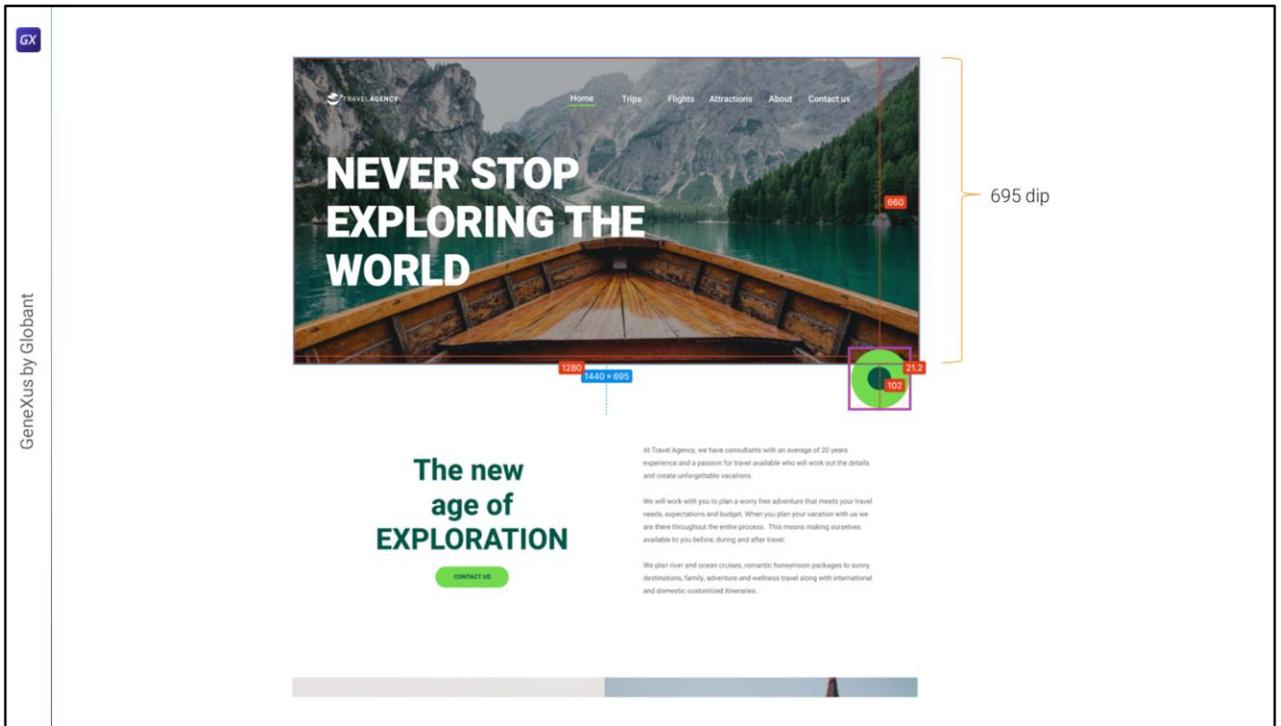
Height

Bottom

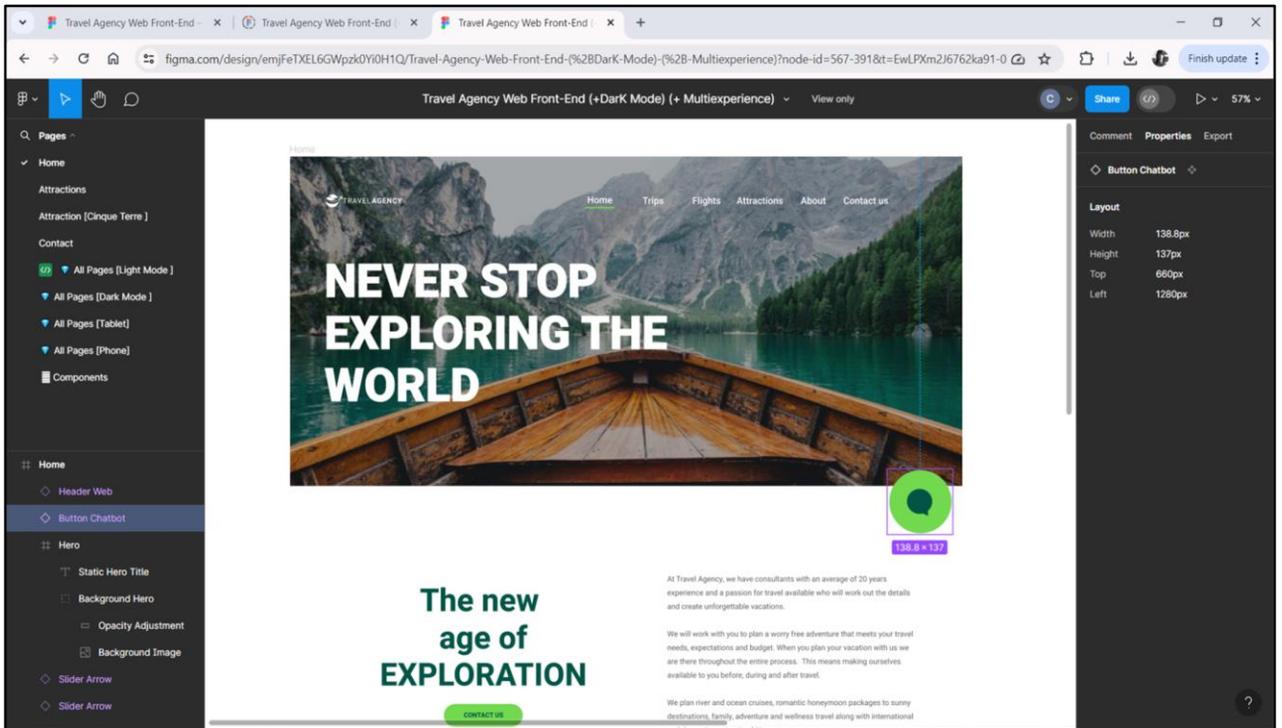
<ContentPlaceholder>

El mismo tipo de análisis vale para el otro eje, el de las y.

Así tendremos **Top** para la distancia del borde superior del canvas, **Height** para el alto del control y **Bottom** para la distancia respecto al borde inferior.



Si buscamos las dimensiones en Figma, vemos que la imagen ocupará de alto 695 píxeles o dips, y que el botón iniciará verticalmente a 660 dips o píxeles del borde superior del canvas.



Por otro lado vemos que el alto del botón será de 137 píxeles o dips...

GeneXus by Globant

The screenshot shows a website layout with the following dimensions and annotations:

- Header height: 695
- Button height: 102
- Text height: 137
- Text width: 239
- Text top offset: 660
- Text right offset: 21
- Text bottom offset: 102

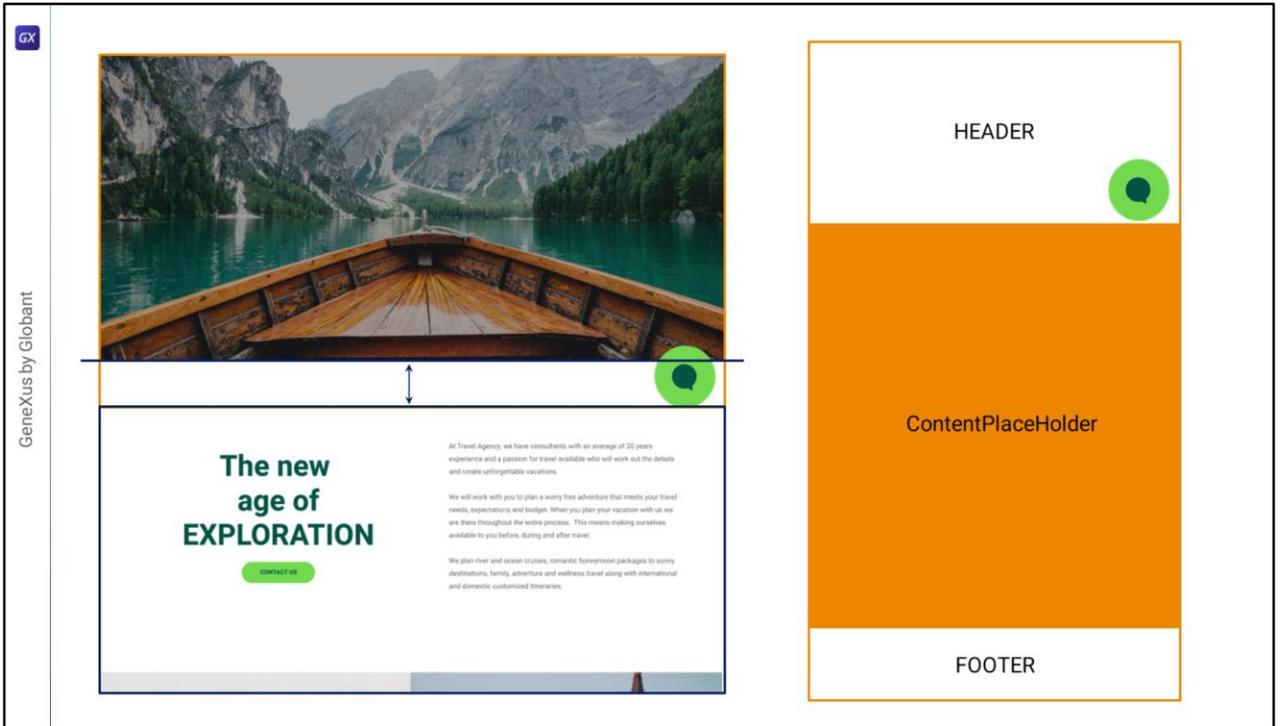
The table below shows the absolute position of the text element:

Absolute position	
Top	660dip
Left	100%
Bottom	100%
Right	21dip
Width	239dip
Height	137dip
Z-Order	1

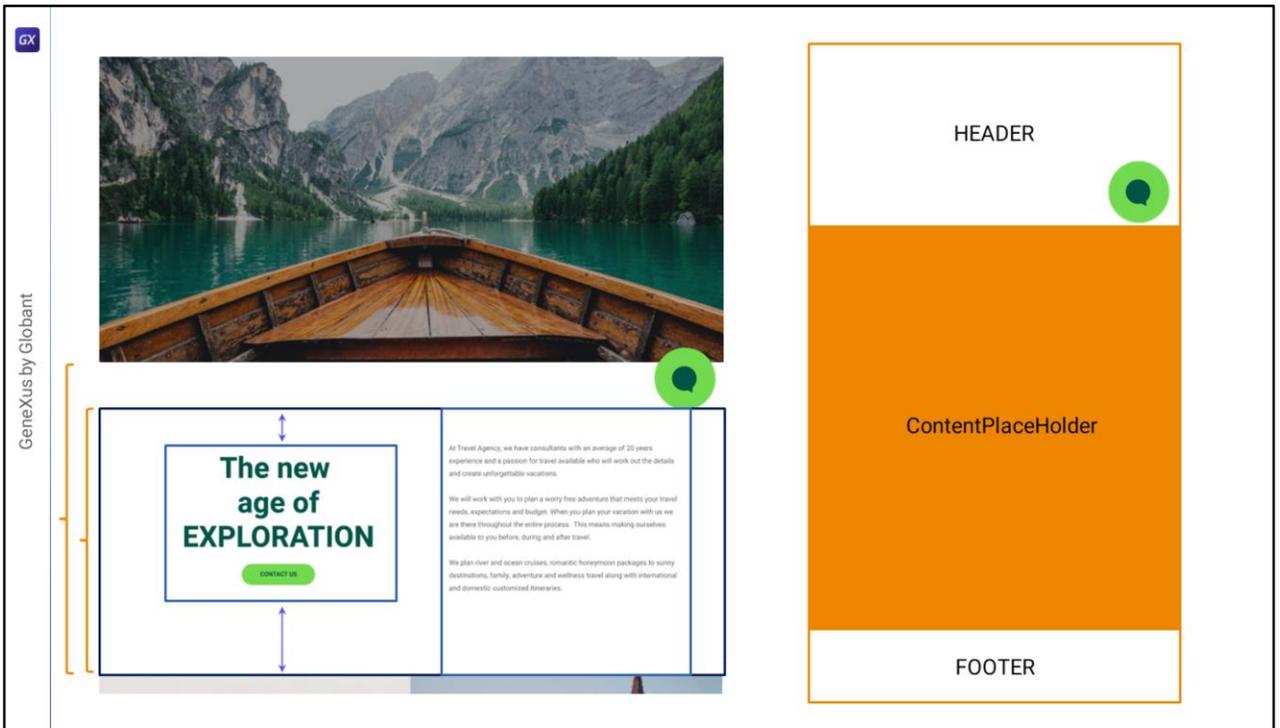
Entonces así definiremos las propiedades: **Top**: 660, **Height**: 137, y como sabemos, fijando 2 la tercera, **Bottom**, quedará necesariamente de 100%. ¿100% de qué? De restar del alto del Canvas, que aún no dijimos cuál es, los 660 del Top y los 137 del Height.

¿Cuál será el alto del Canvas? Debemos especificarlo. Podríamos pensar en darle 695 dips de alto al canvas, de modo que el botón desborde su límite. Y entonces ese Bottom termine siendo un número negativo: -102 dips en este caso.

Sin embargo, así coloquemos ese Height de 695 para el Canvas, **lo estirará**, debido a que internamente todo canvas tiene la propiedad **Auto Grow** en **true**, lo que hará que se estire para contener en la dirección vertical a todos sus controles. Por lo que el alto del Canvas terminará siendo de 695 más estos 102. Por lo que ese Bottom de 100% terminará siendo de 0 dips, es decir, el botón terminará verticalmente sobre el borde inferior del canvas. Por tanto independientemente de que le asignemos 695 dips, el alto del canvas terminará en realidad siendo la suma de este alto más este alto, que da 797 dips.



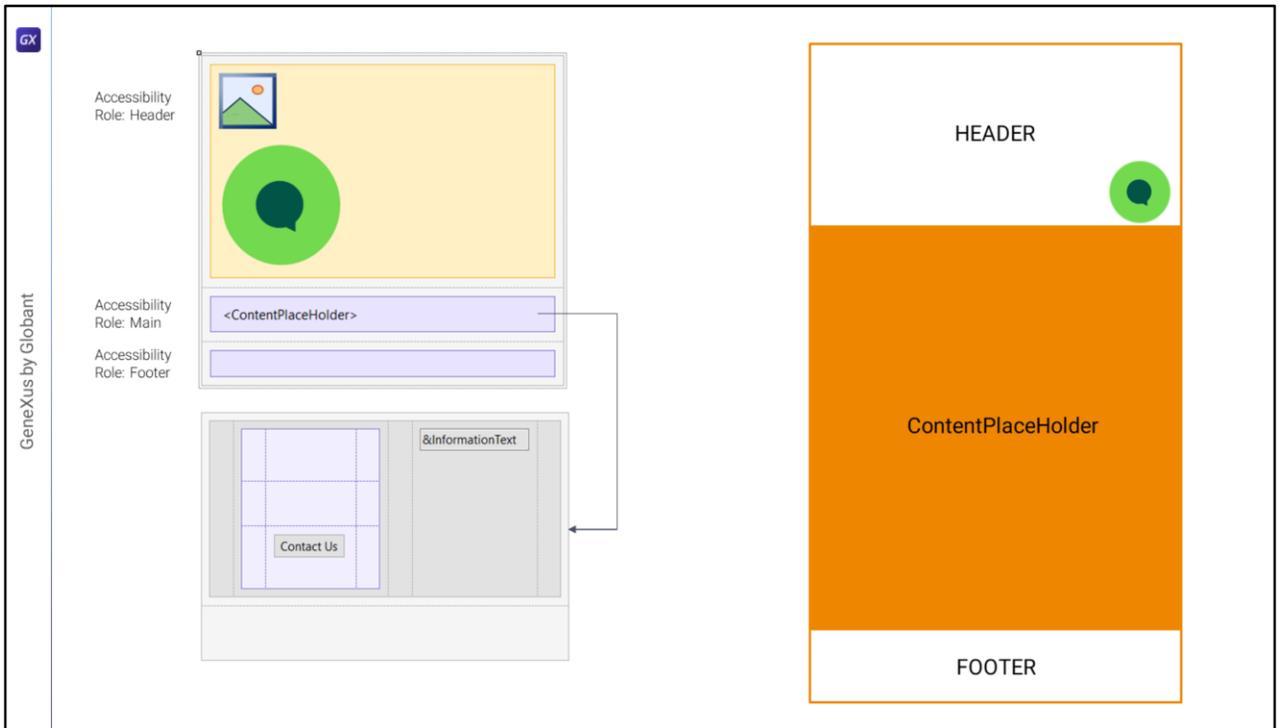
Hasta ahí vamos bien. Pero observemos que esta solución nos trae aparejado un problema. Si el Header se expande para terminar donde termina el botón, entonces deberíamos considerar que el layout de todo panel que se cargue en el ContentPlaceholder iniciará aquí... y no aquí.



Entonces, si queremos mantener esta estructura del Master Panel tendremos que modificar la implementación de esta sección, cambiando el alto de esta tabla para que ahora sea este, en lugar de este...

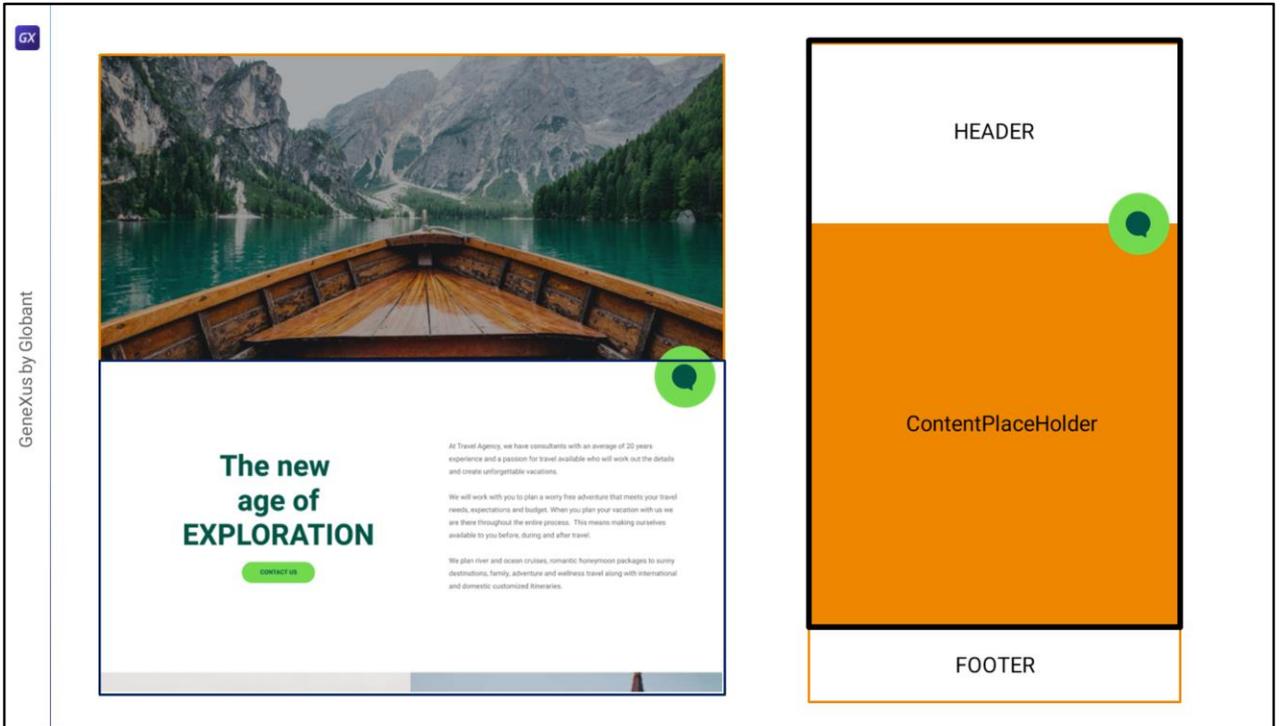
... y la alineación vertical que habíamos dado a esta otra tabla, ya que ahora no podrá estar más en el medio...

Y también tendremos que cambiar el margen superior e inferior que le habíamos proporcionado a este textblock.

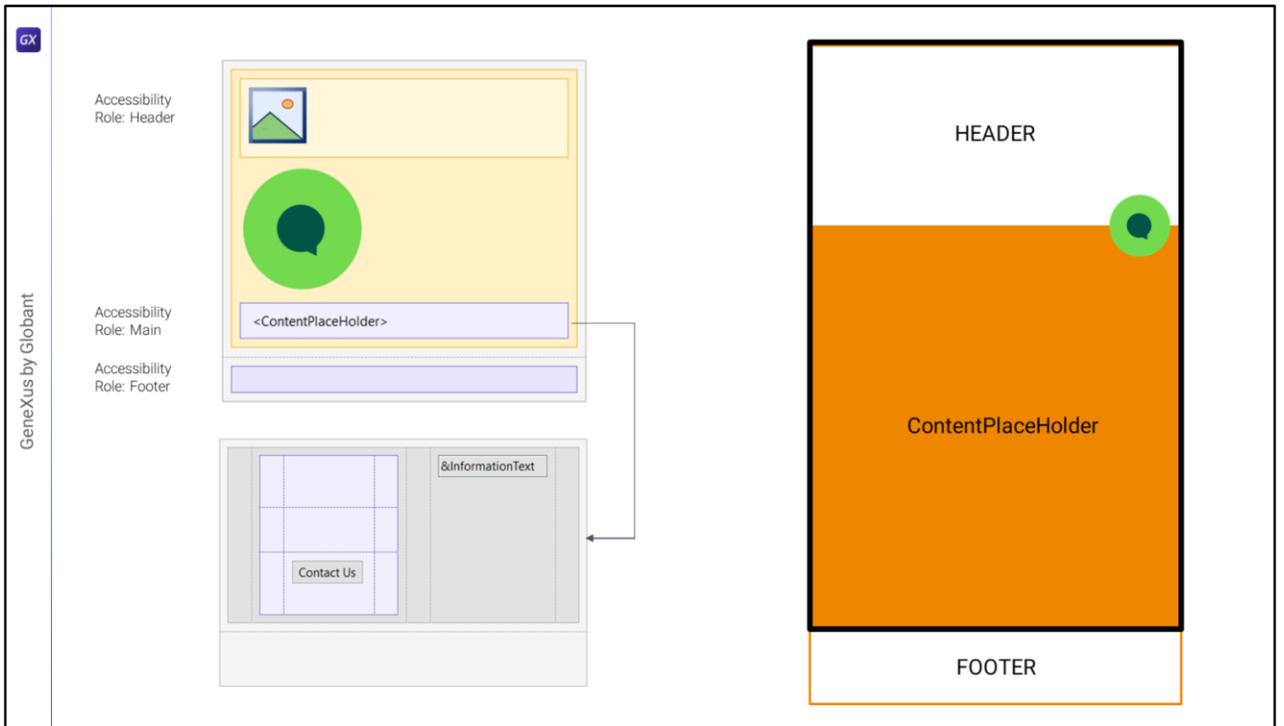


Entonces más o menos así veremos esta implementación del layout abstracto del Master Panel: con una tabla main de 3 filas: en la primera, de Accessible Role Header, un canvas, en la segunda una tabla con Accessible Role Main con el content placeholder y la tercera para el Footer...

Y aquí vemos la implementación del panel Home o el de Attractions... donde tendremos que cambiar la manera en que ubicábamos verticalmente estos controles y el alto de esta tabla...



La otra alternativa, que no requeriría hacer estos cambios, y que es más apropiada conceptualmente, sería considerar que dado que en verdad hay un triple solapamiento: de la sección del Header, el botón, y el ContentPlaceholder, en realidad deberíamos cambiar la estructura del Master Panel para utilizar un Canvas para modelar el triple solapamiento.



Así nos quedaría el layout abstracto del Master Panel: la tabla main, ahora con dos filas y no tres: en la primera un canvas que contiene otro canvas con la propiedad de accesibilidad Header (para eso lo necesito, para poder asignarle propiedad Role, y porque aquí también se superpondrán el menú, el logo y el texto, lo veremos luego), el botón, y la tabla con el ContentPlaceholder y propiedad Main para el Role. Y en la segunda fila de la tabla Main va el Footer.

En este caso no debemos modificar absolutamente nada del layout del panel que se cargará en el ContentPlaceholder, porque comienza aquí.

En el siguiente video implementaremos todo esto en GeneXus

GX

GeneXus by Globant

GeneXus[™]
by Globant

training.genexus.com