

Multiple layouts in a Grid



Rodolfo Roballo

Anteriormente vimos que era posible asignar distintos layouts a un panel. En este video veremos que también es posible asignar distintos layouts a las filas de un grid para que varíen su apariencia dependiendo de su contenido o cambiar el diseño al hacer tap sobre una fila en particular.

GeneXus by Globant

The screenshot displays the GeneXus IDE interface. On the left, three data models are shown: 'Customer' with attributes like CustomerId, CustomerName, CustomerAddress, CustomerPhone, CustomerEmail, CustomerPhoto, CustomerTotalPurchases, CountryId, and CountryName; 'Country' with CountryId, CountryName, CountryFlag, City, and CityName; and 'Invoice' with InvoiceId, InvoiceDate, CustomerId, CustomerName, and InvoiceTotalAmount. On the right, a code snippet for 'InvoiceCollection' is shown, containing three 'Invoice' objects with specific dates and customer information. Below these are three UI layout diagrams: 'Standard customer' with fields for Photo, Name, Country, and Total \$; 'VIP customer' with fields for Photo (marked with a red 'V.I.P.' stamp), Name, Country, and Total \$ (in red); and 'Customer more info' with fields for Photo, Name, Country, Email, Phone, and Address.

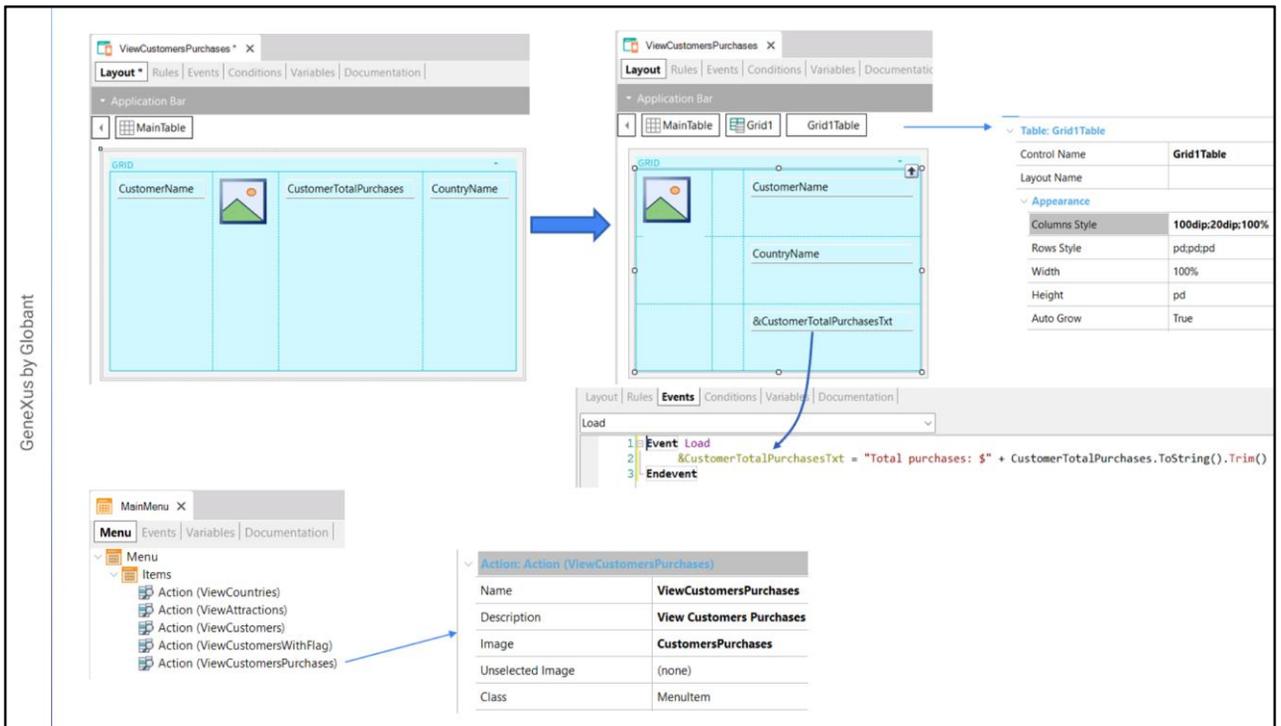
Vamos a partir de un modelo de transacciones en el que contamos con una transacción de clientes, una de países y una de facturas. La transacción de facturas no tiene líneas, porque en este momento nos importa solamente el total de la factura y los datos del cliente al que se factura. También tenemos creado el Invoice_DataProvider con los datos de las facturas de cada cliente.

El total facturado a cada cliente se calcula como la suma de sus facturas, mediante el atributo fórmula CustomerTotalPurchases.

Si quiere contar con estos objetos, importe el archivo invoices.xpz que se encuentra en la sección Materiales de la página del curso.

Queremos ver la lista de clientes, con el nombre del cliente, su foto, el país al que pertenece y el total facturado hasta el momento. Pero queremos una funcionalidad particular y es que a aquellos clientes cuyo total facturado supere los \$5000, deben aparecer el total en negrita y en otro color y además una imagen superpuesta a la foto que informe de que se trata de un cliente VIP, mientras que el resto de los clientes deben aparecer sin estos destacados.

Es decir, vamos a variar el contenido y apariencia de la información de cada fila, en función de la información que vamos a mostrar. Además queremos que cuando hagamos tap en cualquier cliente, se amplíe la información del mismo, mostrándose además el email, el teléfono y la dirección, pero sin el total de compras.



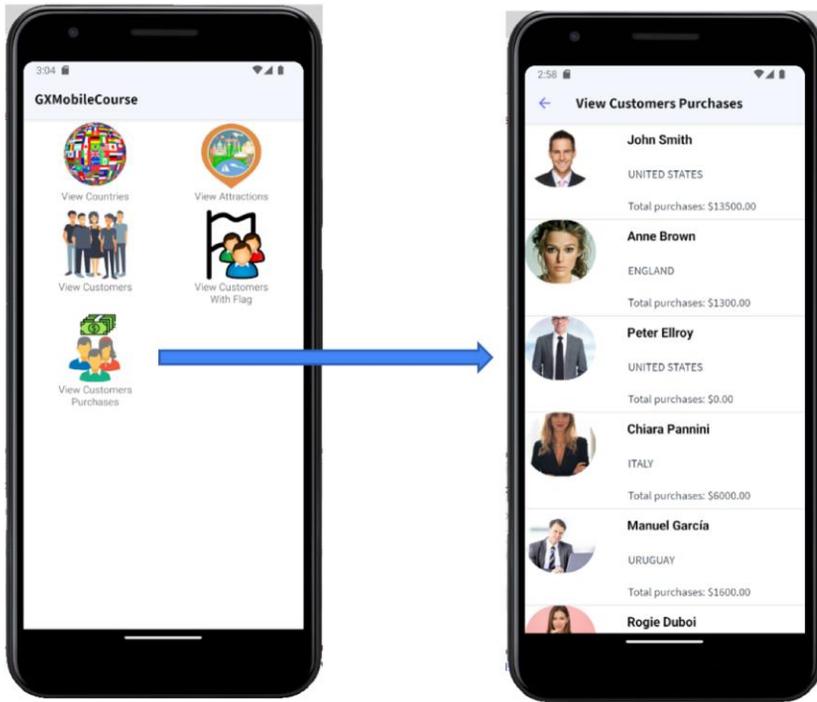
Comenzamos creando un panel `ViewCustomersPurchases` y arrastrando un control grid, con los atributos `CustomerName`, `CustomerPhoto` y `CountryName`. También Creamos una variable `&CustomerTotalPurchasesTxt` del tipo `Varchar(40)` y la agregamos al grid.

Movemos los atributos para llegar al diseño que nos habíamos planteado, con la foto a la izquierda y el resto de los datos a la derecha. Agregamos una columna en el medio como separador. A los atributos de más a la derecha les asignamos `Horizontal Alignment` en `Left` y `Vertical Alignment` en `Middle`. Ahora distribuimos el espacio de las columnas, asignando en la tabla `Grid1Table` la propiedad `Columns Style` los valores: `100 dip, 20 dip y 100%`.

Al atributo `CustomerPhoto` le asignamos en su propiedad `Row Span` el valor `2` para que ocupe 2 filas de alto y le asignamos la clase `ImageRound` que teníamos previamente definida en el `Design System` creado. Y al atributo `CustomerName` le asignamos la propiedad `Class` con la clase `AttributeBold`.

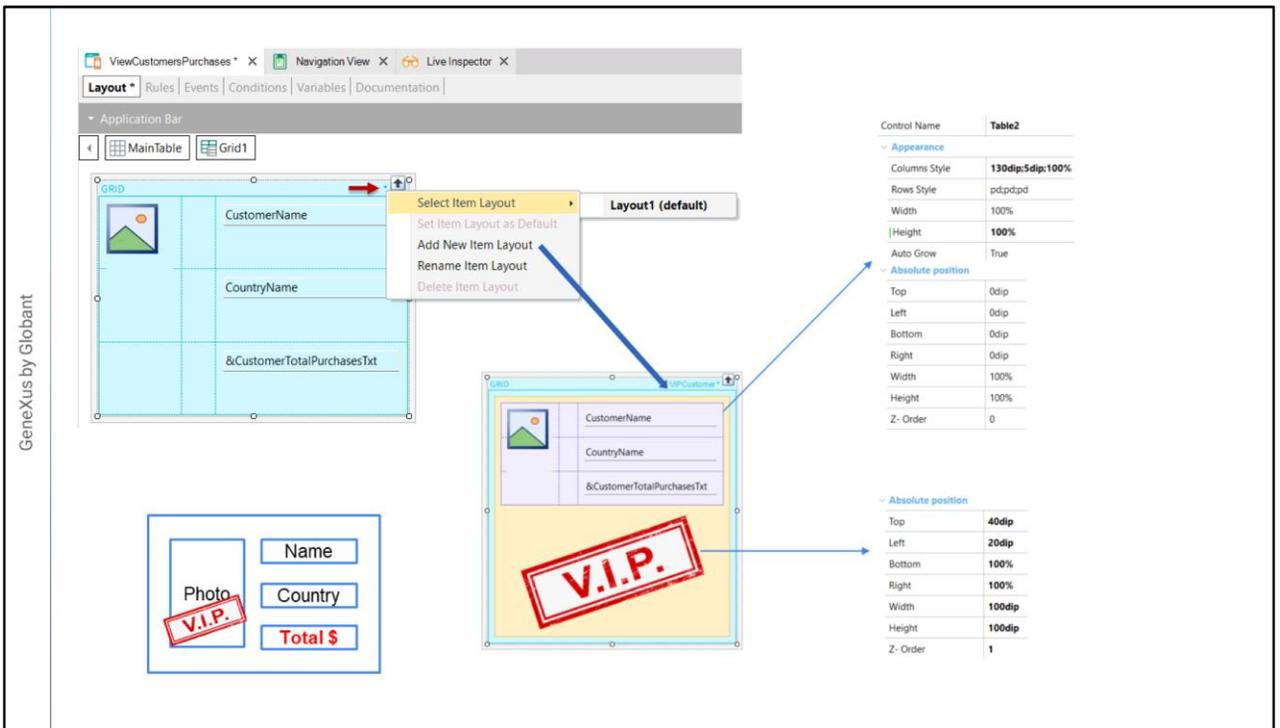
En la solapa de `Eventos`, seleccionamos al evento `Load` y asignamos la variable `&CustomerTotalPurchasesTxt` con el texto "Total purchases: \$" y le concatenamos al atributo `CustomerTotalPurchases` convertido a `string` y sin espacios.

Abrimos el menú `MainMenú` y arrastramos el panel `ViewCustomersPurchases` al nodo `Items`. Y luego asignamos en el action que se creó, asignamos la propiedad `Image` con la imagen `CustomersPurchases`. Ejecutamos el `Android Studio`, abrimos el emulador que teníamos creado. Y presionamos `F5`.



Hacemos tap sobre el icono View Customers Purchases y vemos la lista de clientes, con sus datos personales y los importes facturados, de acuerdo a lo que esperábamos ver.

Ahora vamos a cumplir con el primer requerimiento que determinaba que aquellos clientes que superaran los \$5000 se vieran destacados, con un diseño diferente que los clientes que no habían llegado a esa cifra.



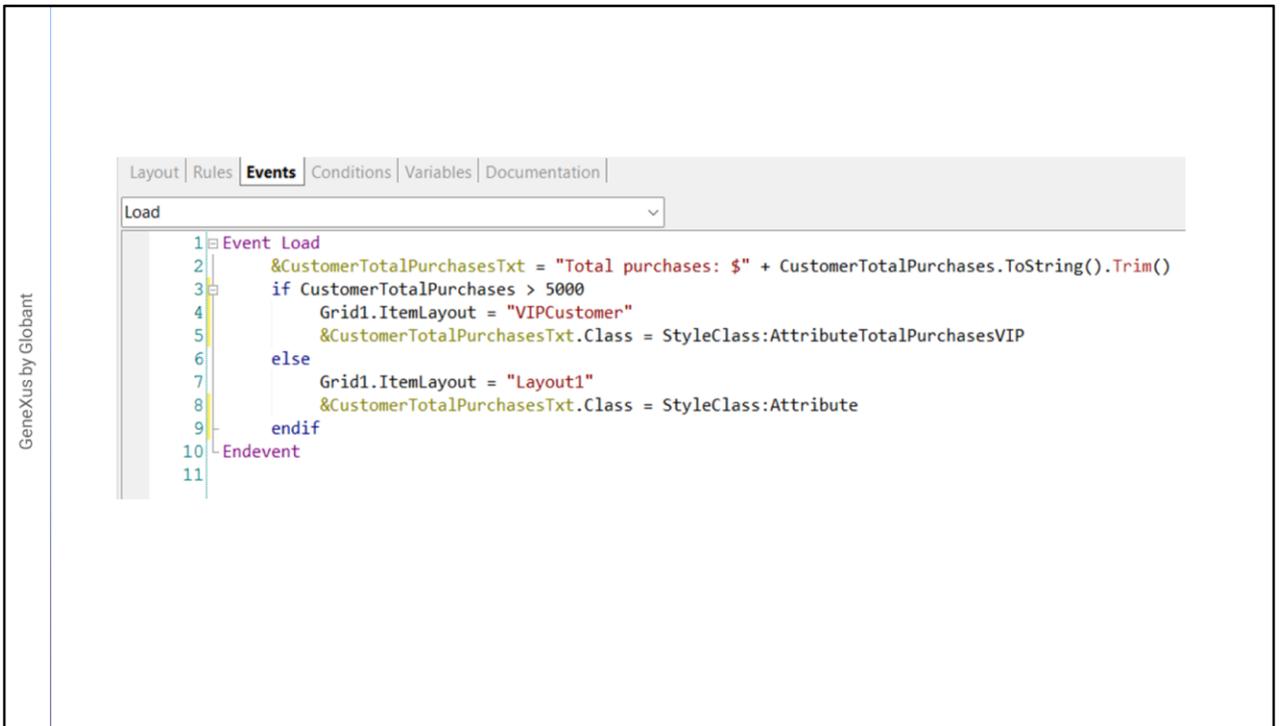
Para hacer que el diseño de cada fila cambie de acuerdo a lo que necesitemos, debemos crear un nuevo layout para el grid.

Si presionamos la flechita de color verde que está arriba a la derecha del grid, vemos que se abre un menú que nos informa que el layout seleccionado por defecto es el Layout1 y que también podemos renombrarlo o crear un layout nuevo.

Vamos a seleccionar Add New Item Layout y le damos de nombre VIPCustomer. Vemos que aparece el nombre arriba en la esquina superior derecha y que el nuevo layout se inicializa con el diseño del layout por defecto, pero podremos hacer los cambios que queramos que ambos layouts quedarán completamente independientes.

Para hacer que el sello de VIP quede por encima de la foto, necesitamos un control Canvas, así que empezamos quitando a todos los controles del grid...arrastramos un control Canvas al grid y luego arrastramos adentro del canvas una tabla, donde volvemos a poner a los atributos con el diseño que teníamos. A la tabla vamos a ajustar los valores de Column Style dándole a la foto 130dip, a la columna separadora 5 dip y luego a la tercer columna, la de los datos, un 100%. A las propiedades de Absolute position las dejamos por defecto, por lo que el Z-Order quedará en 0 y esta grilla estará en la capa de más debajo del layout.

Debajo de la tabla arrastramos una imagen, y la asociamos a la imagen VIPCustomer; una imagen con fondo transparente, que teníamos en la KB. En las propiedades de la Sección Absolute position asignamos Top en 40dip, Left en 20dip, Bottom y Right en 100% y Width y Height en 100dips y Z-Order en 1 para que el sello de VIP quede por encima de la foto del cliente.



Ahora en la sección de eventos agregamos el siguiente código de modo que si el total de compras del cliente supera los \$5000, asignamos a la grilla el layout VIPCustomer y de lo contrario asignamos el Layout1.

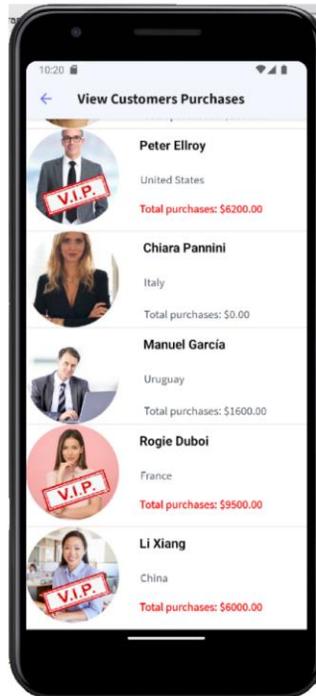
También aprovecharemos la bifurcación lógica para asignar una clase a un control en tiempo de ejecución, de modo que el formato del total de compras dependa también de si el total de compras supera el requisito para el cliente VIP o no.

Primero abrimos el objeto Design System GXMobileCourseDSO y creamos una clase AttributeTotalPurchasesVIP que herede de AttributeBold y le definimos un tamaño de fuente de 16 pixels y color rojo. Y salvamos.

Ahora volvemos a los eventos del panel y donde detectamos que el cliente es VIP, escribimos la variable &CustomerTotalPurchasesTxt, punto Class y utilizando StyleClass asignamos la clase AttributeTotalPurchasesVIP. Y en el else asignamos la clase Attribute.

De esta manera el total de compras se verá resaltado en color rojo, solamente para aquellos clientes que son VIP.

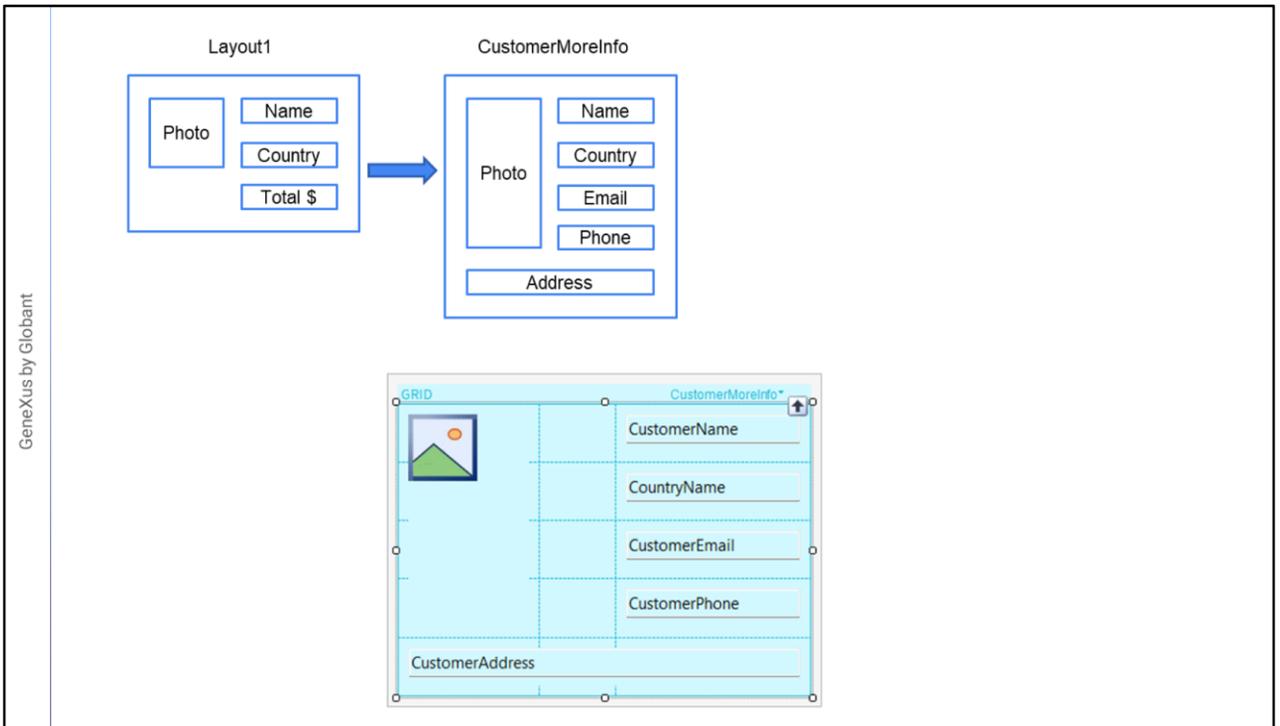
Damos F5...



Hacemos tap sobre el icono View Customers Invoices y vemos que algunos clientes tienen el sello de cliente VIP y otros no. Verificamos que los que tienen el sello son los que superan los \$5000 en el total de compras como queríamos y que este total para esos clientes sale resaltado en rojo.

Por ejemplo Peter Ellroy que llegó a los 6200 es VIP y si vamos más para abajo vemos que también Rogie y Li son clientes VIP.

Los otros clientes que no alcanzaron este total, tienen los datos sin ningún destaque, es decir que dependiendo de una condición fuimos capaces de que algunas filas tengan un formato distinto a otras y este formato se cambió en ejecución dependiendo de los datos.



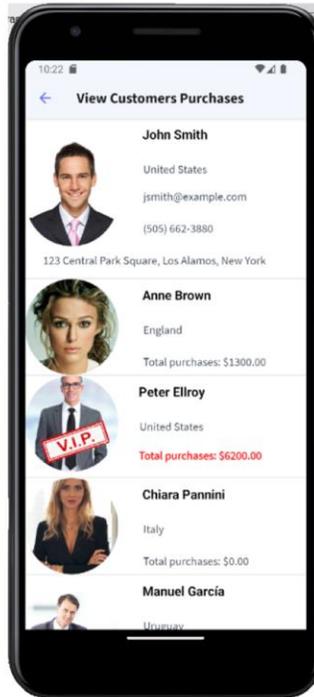
Ahora vamos a cumplir el último requerimiento que cuando hagamos tap en un cliente, se muestre su información ampliada.

Queremos que cuando hagamos tap en una fila cambie el diseño de los datos del cliente y se muestren además de su nombre y su país, su email, su teléfono y su dirección, pero sin que se muestre el total de facturas.

Para esto tenemos que crear un nuevo layout para el grid, basado en el Layout1 que es el layout por defecto, así que primero nos posicionamos en ese layout y luego presionamos la flecha verde, seleccionamos Create new layout, y le damos el nombre CustomerMoreInfo.

Ahora borramos al atributo CustomerTotalPurchases y agregamos a los atributos CustomerEmail, CustomerPhone y CustomerAddress...Ajustamos el diseño del panel...alineamos todos los campos justificando horizontalmente a la izquierda y con VerticalAlignement en Middle.. Y al CustomerAddress le asignamos ColSpan en 3.. y salvamos.

Para hacer que este layout sea utilizado cuando hacemos tap sobre una fila, vamos a las propiedades del grid y en la propiedad Default Select Item Layout seleccionamos al layout CustomerMoreInfo. Damos nuevamente F5.



Abrimos View Customer Purchases y ahora damos tap en el primer cliente.. Y vemos que cambió su layout mostrándose todos los datos patronímicos del cliente.

Si hacemos tap nuevamente volvemos al diseño por defecto. Probamos también lo mismo en un cliente VIP y comprobamos que también se muestran los datos completos del cliente.



En este video vimos cómo asignar distintos layouts a un grid en tiempo de diseño y en tiempo de ejecución, logrando una mayor riqueza en la forma de mostrar datos en la interfase de usuario.

En siguientes videos seguiremos trabajando con otros controles de pantalla que nos provean funcionalidad y estética a nuestra aplicación.