

Multiple layouts in a Grid



Rodolfo Roballo

Anteriormente vimos que era possível atribuir diferentes layouts a um panel. Neste vídeo veremos que também é possível atribuir diferentes layouts às linhas de um grid para que variem sua aparência dependendo de seu conteúdo ou alterar o design ao tocar sobre uma linha em 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, CityId, and CityName; and 'Invoice' with InvoiceId, InvoiceDate, CustomerId, CustomerName, and InvoiceTotalAmount. On the right, a code snippet for 'InvoiceCollection' is visible, showing three invoice objects with specific dates and customer names (Peter Ellroy, Anne Brown, John Smith) and their total amounts (3700, 500, 300). Below these are three UI layout templates: 'Standard customer' with fields for Photo, Name, Country, and Total \$; 'VIP customer' with a red 'V.I.P.' stamp over the Photo field, and Name, Country, and Total \$ (where Total \$ is bold and red); and 'Customer more info' with fields for Photo, Name, Country, Email, Phone, and Address.

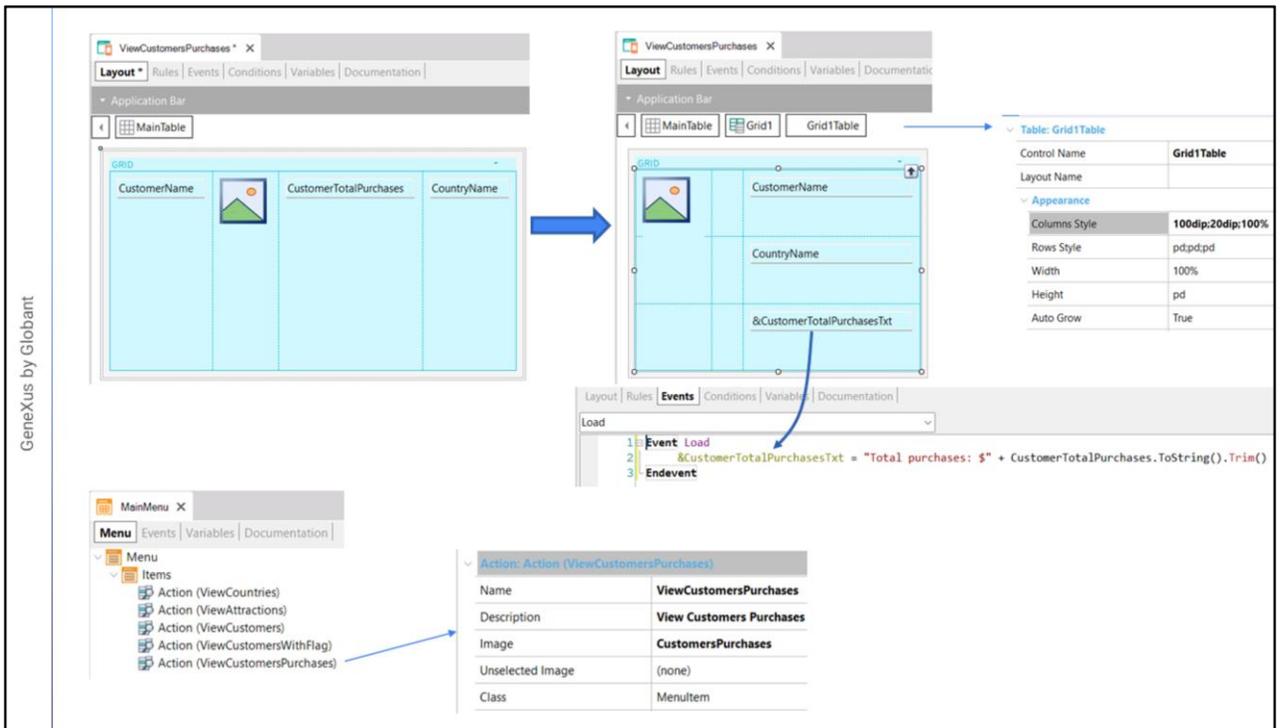
Vamos partir de um modelo de transações em que temos uma transação de clientes, uma de países e uma de faturas. A transação de faturas não possui linhas, pois neste momento nos interessa somente o total da fatura e os dados do cliente a quem é faturado. Também criamos o Invoice_DataProvider com os dados das faturas de cada cliente.

O total faturado a cada cliente é calculado como a soma de suas faturas, usando o atributo fórmula CustomerTotalPurchases.

Se quiser ter estes objetos, importe o arquivo invoices.xpz que se encontra na seção Materiais da página do curso.

Queremos ver a lista de clientes, com o nome do cliente, sua foto, o país a que pertence e o total faturado até o momento. Mas queremos uma funcionalidade particular e é que para aqueles clientes cujo total faturado ultrapasse os \$5000, deve aparecer o total em negrito e em outra cor e também uma imagem sobreposta na foto que informe que se trata de um cliente VIP, enquanto o restante dos clientes deve aparecer sem esses destaques.

Ou seja, vamos variar o conteúdo e a aparência de cada linha, dependendo da informação que vamos exibir. Além disso, queremos que ao tocar sobre qualquer cliente, seja ampliada sua informação, mostrando também o e-mail, o telefone e o endereço, mas sem o total de compras.



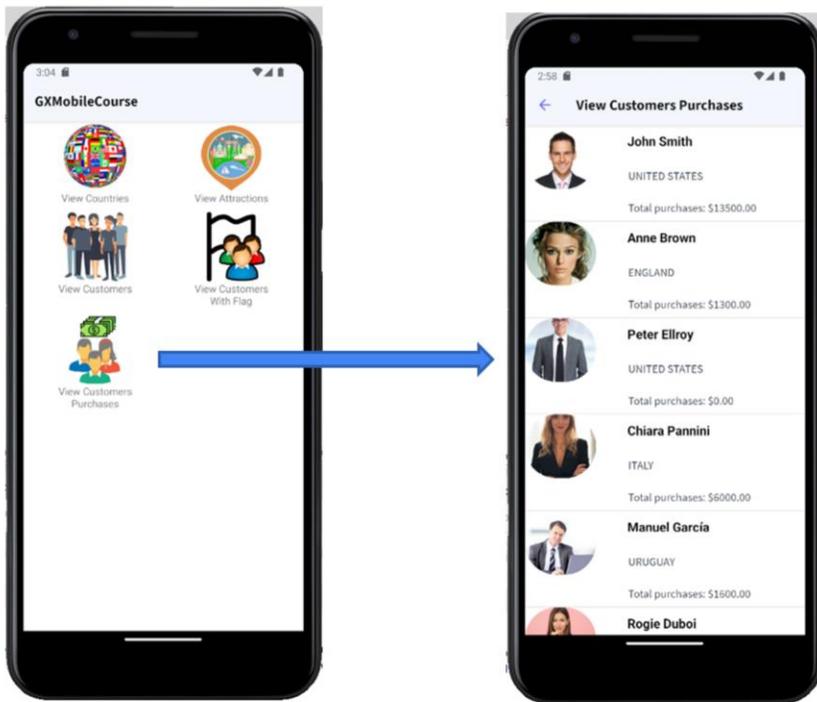
Começamos criando um panel ViewCustomersPurchases e arrastando um controle grid, com os atributos CustomerName, CustomerPhoto e CountryName. Também criamos uma variável &CustomerTotalPurchasesTxt do tipo Varchar(40) e a adicionamos ao grid.

Movemos os atributos para chegar ao design que nos tínhamos proposto, com a foto à esquerda e o restante dos dados à direita. Adicionamos uma coluna no meio como separador. Para os atributos mais à direita, atribuímos Horizontal Alignment em Left e Vertical Alignment em Middle. Agora distribuímos o espaço das colunas, atribuindo na tabela Grid1Table para a propriedade Columns Style os valores: 100 dip, 20 dip e 100%.

Ao atributo CustomerPhoto atribuímos em sua propriedade Row Span o valor 2 para que ocupe 2 linhas de altura e atribuímos a ele a classe ImageRound que tínhamos anteriormente definida no Design System criado. E ao atributo CustomerName atribuímos a propriedade Class com a classe AttributeBold.

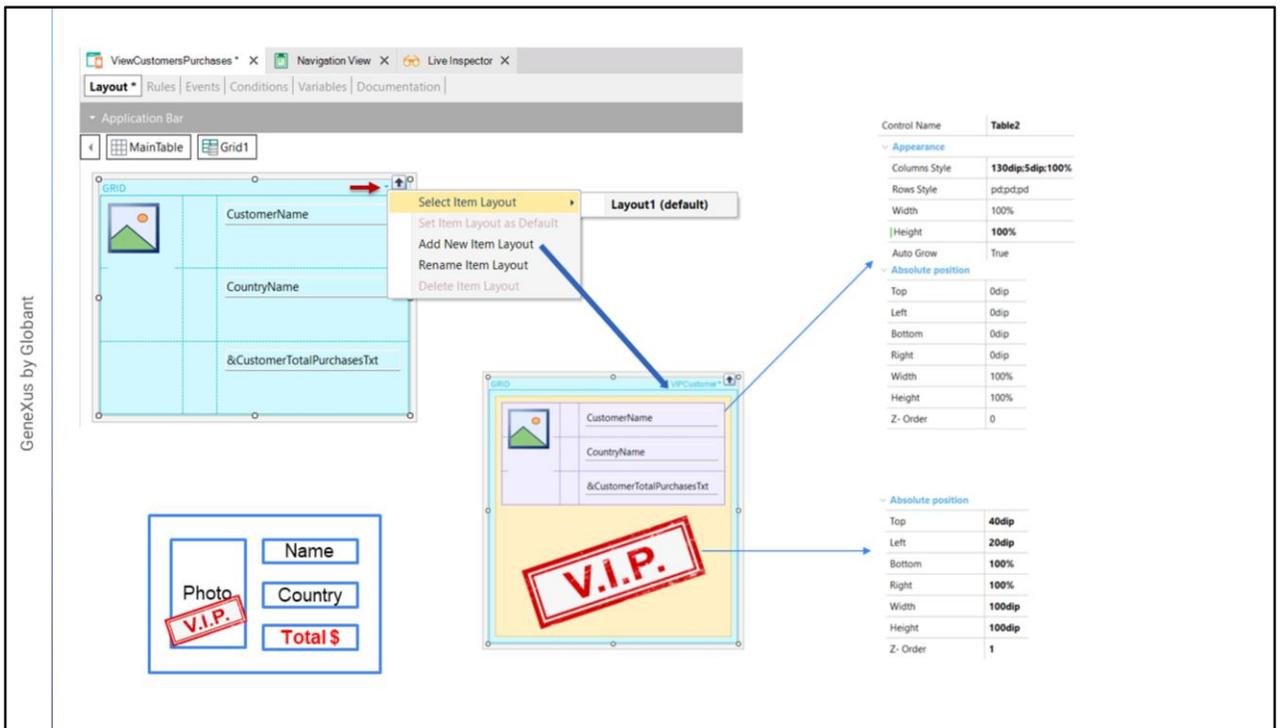
Na aba de Eventos, selecionamos o evento Load e atribuímos a variável &CustomerTotalPurchasesTxt com o texto "Total purchases: \$" e concatenamos ao atributo CustomerTotalPurchases convertido para string e sem espaços.

Abrimos o menu MainMenu e arrastamos o panel ViewCustomersPurchases para o nó Items. E então atribuímos na action que foi criada, a propriedade Image com a imagem CustomersPurchases. Executamos o Android Studio e abrimos o emulador que criamos. E pressionamos F5.



Tocamos sobre ícone View Customers Purchases e vemos a lista de clientes, com os seus dados pessoais e os valores faturados, de acordo com o que esperávamos ver.

Agora vamos cumprir o primeiro requisito que determinava que aqueles clientes que ultrapassassem os \$5000 aparecessem destacados, com um design diferente dos clientes que não atingiram esse valor.



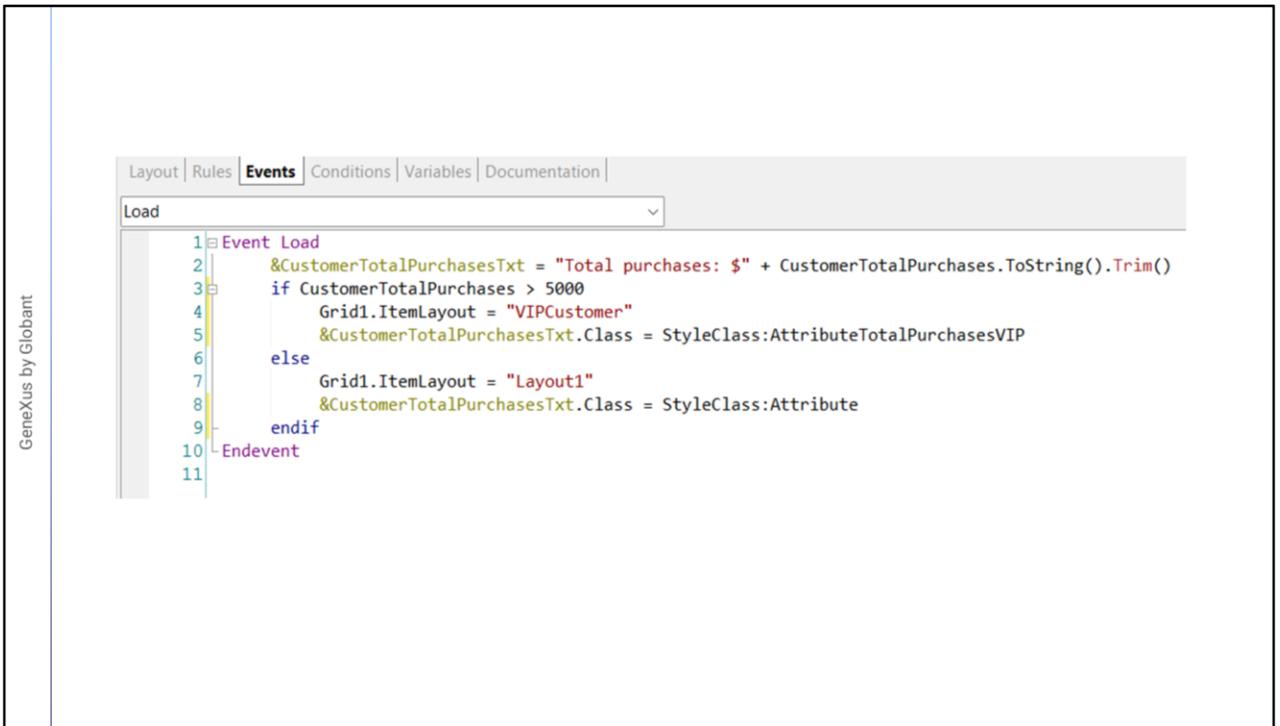
Para fazer com que o design de cada linha mude de acordo com o que necessitamos, devemos criar um novo layout para o grid.

Se pressionarmos a pequena seta verde que está acima à direita do grid, vemos que se abre um menu que nos informa que o layout selecionado por padrão é o Layout1 e que também podemos renomeá-lo ou criar um novo layout.

Vamos selecionar Add New Item Layout e nomeá-lo VIPCustomer. Vemos que aparece o nome acima no canto superior direito e que o novo layout é inicializado com o design do layout padrão, mas podemos fazer as alterações que desejarmos que ambos os layouts ficarão completamente independentes.

Para fazer com que o carimbo de VIP fique sobre a foto, precisamos de um controle Canvas, então começamos removendo todos os controles do grid... arrastamos um controle Canvas para o grid e depois arrastamos dentro do canvas uma tabela, onde voltamos a colocar os atributos com o design que tínhamos. Para a tabela vamos ajustar os valores de Columns Style dando para a foto 130dip, para a coluna separadora 5 dip e depois para a terceira coluna, a dos dados, 100%. Para as propriedades de Absolute position deixamos como padrão, pelo que o Z-Order ficará em 0 e este grid ficará na camada mais baixa do layout.

Abaixo da tabela arrastamos uma imagem, e a associamos à imagem VIPCustomer; uma imagem com fundo transparente, que tínhamos na KB. Nas propriedades da Seção Absolute position atribuímos Top em 40dip, Left em 20dip, Bottom e Right em 100% e Width e Height em 100dips e Z-Order em 1 para que o carimbo de VIP fique sobre a foto do cliente.



Agora na seção de eventos adicionamos o seguinte código de maneira que se o total de compras do cliente exceder os \$5000, atribuímos ao grid o layout VIPCustomer e caso contrário atribuímos o Layout1.

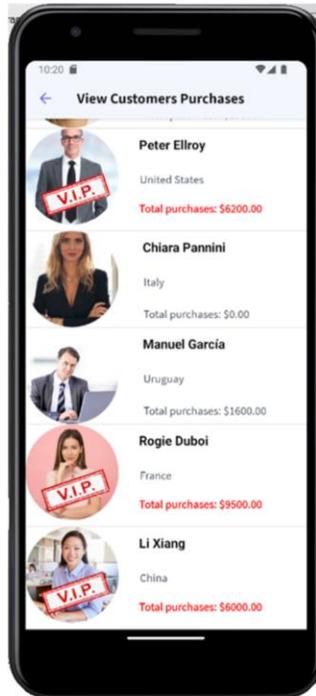
Também aproveitaremos a bifurcação lógica para atribuir uma classe a um controle em tempo de execução, de maneira que o formato do total de compras também dependa se o total de compras excede o requisito para o cliente VIP ou não.

Primeiramente abrimos o objeto Design System GXMobileCourseDSO e criamos uma classe AttributeTotalPurchasesVIP que herda de AttributeBold e definimos para ela um tamanho de fonte de 16 pixels e cor vermelha. E salvamos.

Agora voltamos aos eventos do panel e onde detectamos que o cliente é VIP, escrevemos a variável &CustomerTotalPurchasesTxt, ponto Class e utilizando StyleClass atribuímos a classe AttributeTotalPurchasesVIP. E no else atribuímos a classe Attribute.

Desta forma, o total de compras ficará destacado em vermelho, apenas para aqueles clientes que são VIP.

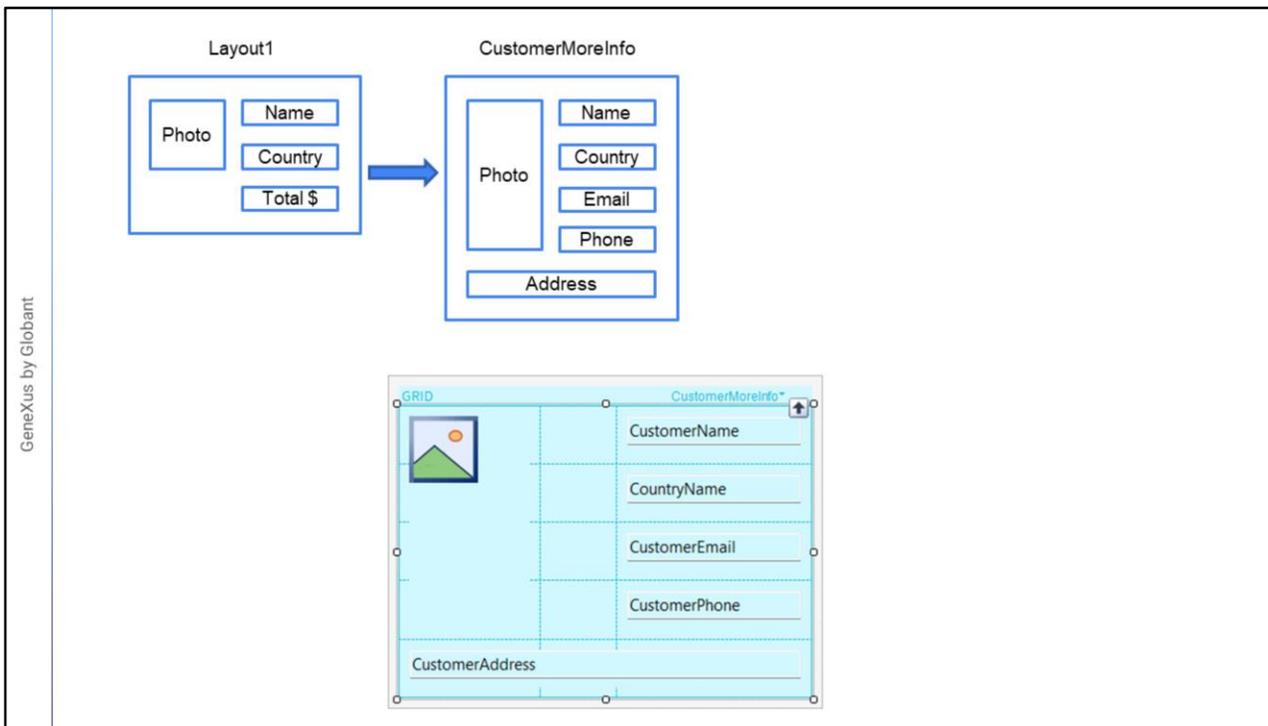
Pressionamos F5...



Tocamos sobre o ícone View Customers Purchases e vemos que alguns clientes possuem o carimbo de cliente VIP e outros não. Verificamos que aqueles que possuem o carimbo são aqueles que ultrapassam os \$5000 no total de compras como queríamos e que este total para esses clientes fica destacado em vermelho.

Por exemplo, Peter Ellroy que atingiu os 6200 é VIP e se descermos mais vemos que Rogie e Li são clientes VIP também.

Os outros clientes que não atingiram esse total possuem os dados sem nenhum destaque, ou seja, dependendo de uma condição fomos capazes de fazer com que algumas linhas tivessem um formato diferente de outras e esse formato foi alterado em execução dependendo dos dados.



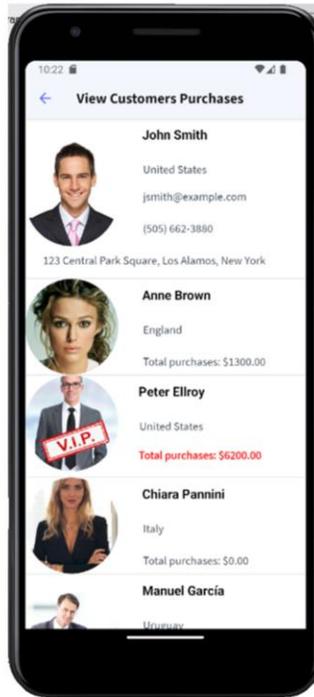
Agora vamos cumprir o último requisito de que ao tocar sobre um cliente, seja exibida sua informação ampliada.

Queremos que, ao tocar sobre uma linha, altere o design dos dados do cliente e sejam exibidos, além de seu nome e seu país, seu e-mail, seu telefone e seu endereço, mas sem que seja apresentado o total de faturas.

Para isso temos que criar um novo layout para o grid, baseado no Layout1, que é o layout padrão, então primeiro nos posicionamos nesse layout e depois pressionamos a seta verde, selecionando Create new layout e damos a ele o nome de CustomerMoreInfo.

Agora excluimos o atributo CustomerTotalPurchases e adicionamos os atributos CustomerEmail, CustomerPhone e CustomerAddress... alinhamos todos os campos justificando horizontalmente à esquerda e com VerticalAlignment em Middle... Ajustamos o design do panel... E ao CustomerAddress atribuímos ColSpan em 3... e salvamos.

Para fazer com que este layout seja utilizado quando tocamos sobre uma linha, vamos até as propriedades do grid e na propriedade Default Selected Item Layout selecionamos o layout CustomerMoreInfo. Pressionamos novamente F5.



Abrimos View Customers Purchases e agora tocamos sobre o primeiro cliente.. E vemos que mudou seu layout, mostrando todos os dados próprios do cliente.

Se tocamos novamente, voltaremos ao design padrão. Testamos também o mesmo em um cliente VIP e verificamos que também são exibidos os dados completos do cliente.



Neste vídeo vimos como atribuir diferentes layouts a um grid em tempo de design e em tempo de execução, conseguindo uma maior riqueza na forma de mostrar dados na interface de usuário.

Nos próximos vídeos continuaremos trabalhando com outros controles de tela que proporcionam funcionalidade e estética à nossa aplicação.