

**GeneXus**<sup>™</sup>  
The power of doing.

## Controls: SD Maps, Rating, SD Smart Grids, Switch

User Experience & User Interface

*GeneXus 16*

## Control Types

- 1 SD Maps
- 2 Rating
- 3 SD Smart Grids
- 4 Switch

Continuaremos com o estudo de diferentes tipos de controles que nos ajudarão a fazer com que nossa aplicação tenha uma Experiência de Usuário e uma Interface mais amigável. Começaremos com o controle SD Maps para ver informações em um mapa, depois veremos o controle Rating para atributos numéricos, continuaremos com o tipo de controle SD Smart Grids que nos permitirá visualizar grids com características melhoradas e por último veremos o uso do controle Switch que podemos usar com atributos booleanos.

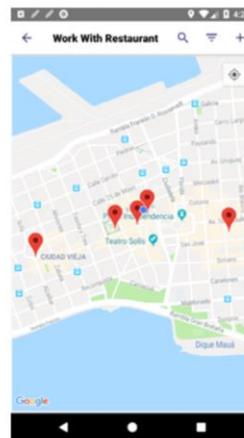
## Control Type SD Maps

SD Maps

### Standard Grid



### SD Maps Grid



O control type é uma propriedade de determinados controles que afeta a maneira como o controle será exibido e utilizado.

Dependendo do tipo escolhido, diferentes propriedades particulares de cada tipo serão exibidas para serem configuradas.

Como já vimos, para oferecer aos participantes do evento uma lista de restaurantes próximos com descontos, foi criada a transação Restaurant. Se quisermos que o grid que implementa a lista de restaurantes não seja mostrado na forma padrão, mas seja mostrado como locais (pins) em um mapa, já que temos a localização de cada restaurante (sua longitude e latitude)

Controls: SD Maps, Rating, SD Smart Grids, Switch GeneXus

## Control Type SD Maps

SD Maps

Layout Rules Events Conditions Variables

Application Bar Insert

MainTable Grid1

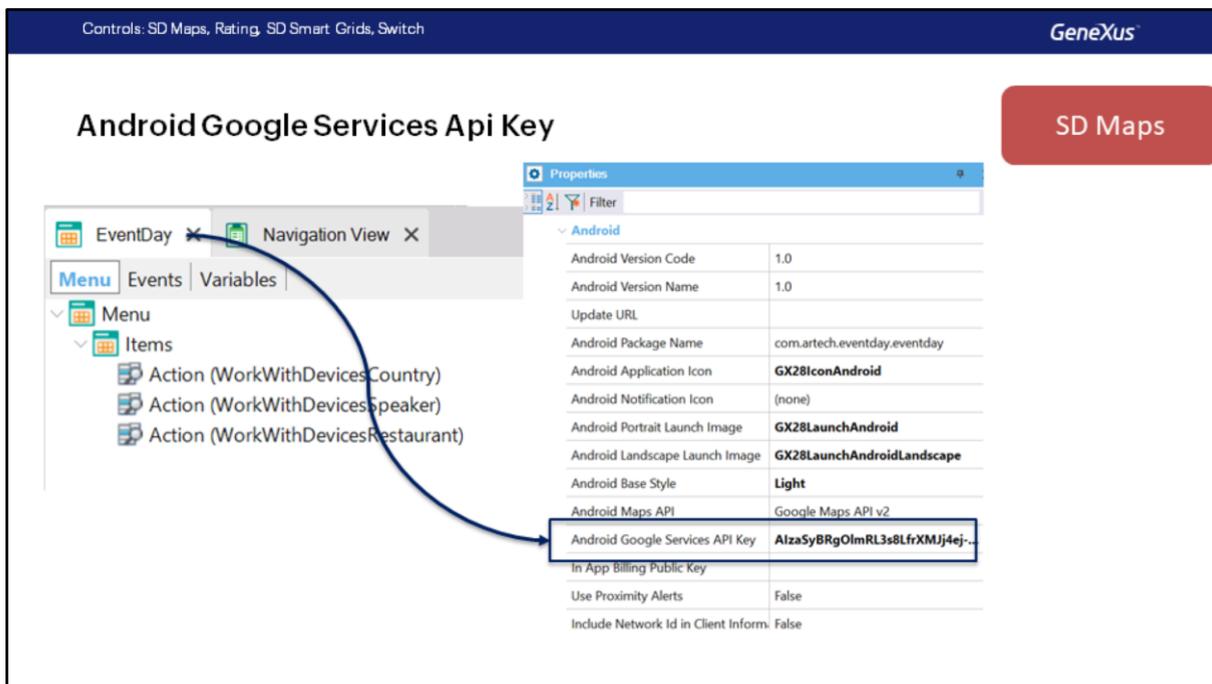
GRID



Control Info

Control Type	SD Maps
Auto Grow	False

Podemos alterar o **Control Type** do grid e escolher, entre todos os SD controls, o **SD Maps**.



Devemos saber que, para poder usá-lo, você deve ter uma API Key do Google que corresponda à assinatura de nossa aplicação. Se você não tiver essa key, os mapas não serão mostrados em execução. Sairá o grid padrão.

Esta chave é configurada no objeto Main, no nosso caso no objeto Menu for Smart Devices EventDay, no grupo de propriedades Android, encontraremos a propriedade Android Google Service Api Key.

A Key é obtida acessando o Developer Console do Google, registrando a app que estamos desenvolvendo. Para ver mais detalhes sobre como obter a API key, você pode visitar nosso wiki onde, com etapas simples, será explicado como fazer isso.

<http://wiki.genexus.com/commwiki/servlet/wiki?19055>

## Control Type SD Maps

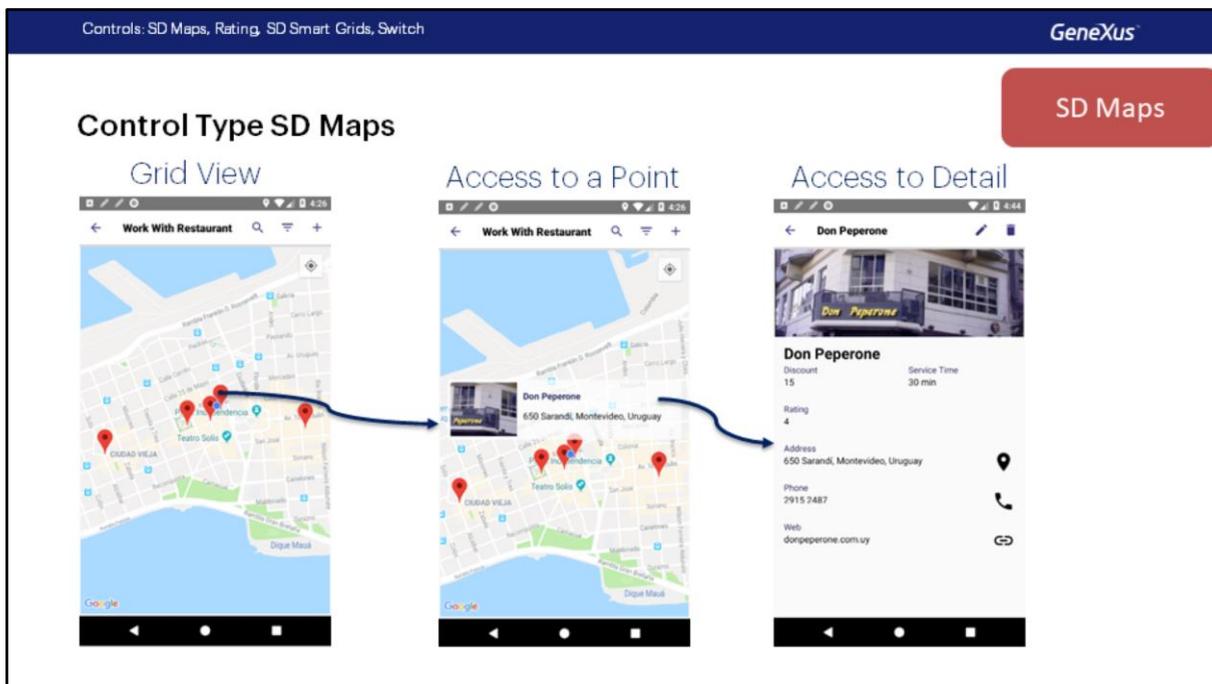
SD Maps

The screenshot displays the GeneXus IDE interface for configuring the SD Maps control. The top menu includes 'Layout', 'Rules', 'Events', 'Conditions', and 'Variables'. Below the menu is the 'Application Bar' with 'MainTable' and 'Grid1' buttons. The central workspace shows a 'GRID' with two text fields: 'RestaurantName' and 'RestaurantAddress'. A blue arrow points from the 'Grid1' button to the 'Control Info' panel on the right. The 'Control Info' panel lists various properties for the 'SD Maps' control, with 'Location Attribute' set to 'RestaurantGeolocation'.

Control Info	
Control Type	SD Maps
Auto Grow	False
Show My Location	True
Map Type	Standard
User Can Choose Map Type	False
Location Attribute	RestaurantGeolocation
Pin Show My Location	locateMe
Pin Image	(none)
Pin Image Attribute	(none)
Pin Image Class	SDMapPinImage
Show Navigation on Full Screen	True
Show Traffic	False
Initial Zoom	Show all points
Center	Default
Selection Layer	False
Directions Layer	False

Devemos inserir de forma obrigatória o nome de um atributo ou variável do grid onde teremos as informações de Geolocalização, pode ser do tipo GeoLocation ou GeoPoint. No nosso caso, usaremos o atributo RestaurantGeolocation.

Para implantá-lo se utilizará la app de mapas del dispositivo. En el caso de Android, Google Maps.



Em tempo de execução, um mapa com todos os pontos será mostrado, no nosso caso, os restaurantes e de forma semelhante ao que acontecia ao fazer Tap em um item do List padrão, ao fazer Tap no pin no mapa, primeiro será mostrado o elemento escolhido, como está especificada a linha no grid e ao fazer tap na linha, será chamada a Default Action (que por padrão será para a tela de Detalhe de um Restaurante no modo View).

## Demo: Using SD Maps

Vejamos como implementamos SD Maps em nossa app.

Primeiro vamos configurar a API Key do Google Maps na aplicação, para isto vamos para as propriedades do Objeto EventDay, o Menu for Smart Devices que é o objeto Main de nossa app. Dentro do grupo de propriedades do Android, encontraremos uma propriedade com o nome Android Google Services API Key.

Anteriormente se utilizava uma propriedade no nível do gerador Android, esta propriedade ficou apenas por compatibilidade com versões anteriores, a propriedade se chama Android Maps Api Key. Nós só utilizaremos a propriedade no nível do Objeto EventDay.

Agora abriremos o objeto WorkWithDevicesRestaurant e vamos para o List.

No Grid, vamos alterar a propriedade Control Type, note que não tem um valor, portanto é mostrado como um grid Standard, alteraremos seu valor para usar SD Maps.

Ao alterar o Control Type, outras propriedades adicionais estarão disponíveis, que serão específicas para cada tipo que escolhemos.

No nosso caso, ao escolher SD Maps, veremos as propriedades

Show My Location: que permite mostrar a localização do dispositivo em tempo real

Map Type: que permite mudar o tipo de mapa usado, pode ser o mapa Standard, pode ser

a vista de Satélite ou pode ser o tipo Híbrido.

A propriedade `User Can Chose Map Type`: quando definido como `True`, permitimos que o usuário altere a visualização.

`Location Attribute`: esta é a única propriedade Obrigatória que devemos configurar, nesta propriedade devemos inserir um nome de atributo ou variável que conterà a localização, deve ser um tipo de dados `GeoLocation` ou `GeoPoint`, no nosso caso iremos inserir aqui o atributo `RestaurantGeoLocation` que é onde armazenamos a longitude e latitude de cada restaurante.

Na propriedade `Pin Show My Location`, colocaremos um ícone para mostrar nossa localização. também temos as propriedades `Pin Image`, `Pin Image Attribute` e `Pin Image Class`, todas elas permitem modificar o ícone das localizações do grid, deixaremos os valores default.

`Show Traffic`: nos permite estabelecer se queremos mostrar o tráfego em tempo real no mapa ou não.

`Initial Zoom`: permite estabelecer o zoom inicial, por padrão, mostrará todos os pontos no mapa, o que significa que o nível de zoom será dinâmico.

`Center`: permite estabelecer onde o mapa será centrado, pode ser em nossa localização ou em outro local determinado. Por padrão, no centro de todos os pontos visíveis.

`Selection Layer`: nos permite selecionar um local em execução, permite mover-se no mapa, tendo um ícone fixo no centro dele, o qual obtém a posição e aciona um evento. Veremos em detalhe o seu uso no nível avançado.

`Directions Layer`: Em execução permite desenhar a rota ou direção entre os pontos do Grid. Cada ponto ou posição no grid é dado pelo atributo ou variável configurado na propriedade `Location Attribute`, também nos permitirá escolher o tipo de transporte que queremos utilizar (por exemplo, de carro ou caminhando). Também veremos seu uso no nível avançado.

Vamos Compilar a aplicação já que à medida que adicionamos um atributo isto modifica um evento `Server Side`, além do valor da propriedade `Google Api Key` ser uma configuração do `Apk`.

Já temos a aplicação, vamos para a lista de Restaurantes, e agora podemos ver que o mapa com os pontos é exibido.

Note que este ícone diferente corresponde à nossa localização que é a Plaza Independencia de Montevideo, como estamos usando um Emulador e este não tem uma posição determinada, é necessário se quisermos “emular” um local, configurá-lo no emulador, para

isso ir para a opção More no lado do emulador, e na Longitude e Latitude podemos definir as coordenadas que queremos emular, por padrão isso não está configurado, depois de configurá-las pressione o botão Send.

Se fazemos Tap em um dos pontos, as informações sobre o Restaurante serão exibidas em uma tabela e, se fazemos Tap novamente nesta tabela, a ação default será executada, neste caso, como será acessada a tela de detalhe de um Restaurante.

Vamos alterar por exemplo o valor de Map Type, retornemos, vejamos como fica, observe que o mapa mudou, agora a camada é mostrada com uma imagem tirada do satélite. Voltemos a deixar a propriedade no valor Standard.

Agora, ativemos a visualização da camada de Tráfego, vamos novamente para o emulador, observe que agora há ruas mostradas em verde, laranja ou vermelho, estas são informações em tempo real fornecidas pelo Google neste caso, sobre o estado do trânsito.

Vamos deixar novamente no valor default.

## Rating Control Type

The screenshot shows the GeneXus IDE interface. On the left, a grid control is displayed with fields for 'RestaurantName' and 'RestaurantAddress', and a 'RestaurantRating' field with a star rating icon. A blue arrow points from the 'RestaurantRating' field in the grid to the 'Control Info' section of the properties panel on the right. The properties panel is titled 'Attribute/Variable: RestaurantRating' and shows various settings. The 'Control Info' section is expanded, showing 'Control Type' set to 'Rating'. Below this, the 'Max Value' is set to 5 and the 'Step' is set to 1. A green button labeled 'Rating' is located in the bottom left corner of the screenshot.

Attribute	RestaurantRating
Label Position	None
Class	Attribute
Visible	True
Invisible Mode	Keep Space
Enabled	True
Format	Combo Box
Invite Message	Radio Button
Link	Edit
Autolink	Check Box
Virtual Keyboard	Dynamic Combo Box
Enter Event	Rating
Control Info	SD Scanner
Control Type	Rating
Max Value	5
Step	1

Rating

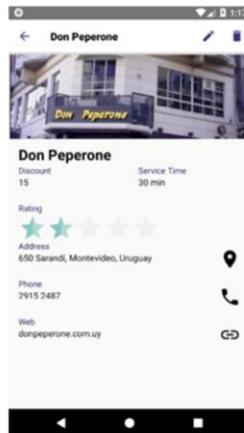
Não apenas para controles grid vale isto que estamos vendo, também os controles do tipo atributo ou variável possuem a propriedade Control type para ser configurada, variando o comportamento default.

Dependendo do tipo de dados de controle, os tipos que teremos disponíveis. Vejamos como podemos fazer para que o usuário insira uma classificação em estrelas, para o atributo RestaurantRating.

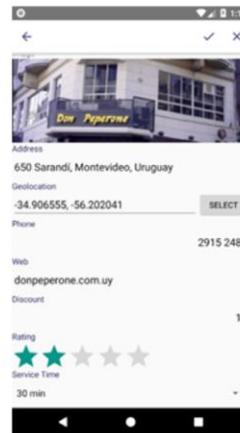
Ao seleccionar como Control Type do atributo o valor Rating, estas duas propriedades serão abertas para indicar o valor máximo de estrelas e se o valor vai de um em um: ou seja, se o usuário escolhe 2 estrelas (ao editar um restaurante), valor armazenado será 2 e o máximo será 5.

## Rating Control Type

## View (Read Only)



## Edit Mode



Rating

Isto será válido para atributos Readonly, como no caso do View de Restaurants. E também se for um campo editável, como no caso do detalhe em modo Edit, onde o usuário selecionará o número de estrelas que deseja.

## Demo: Using Rating Control Type

Vamos ver no GeneXus.

Vamos acessar o `WorkWithDevicesRestaurant` e vamos ao `List`. No meu caso já adiantei um pouco o trabalho no `Grid`, o que fiz foi adicionar o atributo `RestaurantRating` ao grid abaixo de `RestaurantAddress`, também ajustei a imagem de modo que agora tome 3 linhas em vez de 2, a tabela segue como a tínhamos.

Como podem ver à direita no emulador, como agora há mais linhas, a imagem ficou muito grande, lembrem-se que eu estava usando `Fill Keeping Aspect Ratio`, o que eu vou fazer é colocar valores fixos na altura das linhas para que isso não aconteça, vamos inserir em `Rows Style` os valores de `30dip` para a primeira linha, a do nome e `20 dips` para as 2 linhas restantes, a do endereço e a da classificação. Observe que ao inserir um valor numérico assume automaticamente a unidade de medida `DIP`, este é o default.

Vamos alterar o control type por rating para o atributo, deixamos as propriedades de valor máximo e incremento por default.  
vamos gravar e executar a aplicação.

Agora vemos que o valor do atributo é mostrado como estrelas.

Se acessarmos um Restaurante, vemos que no Detalhe o atributo ainda é mostrado como numérico, o mesmo se entrarmos no modo de Edição. Voltemos ao Genexus e vamos mudá-lo nestas duas telas.

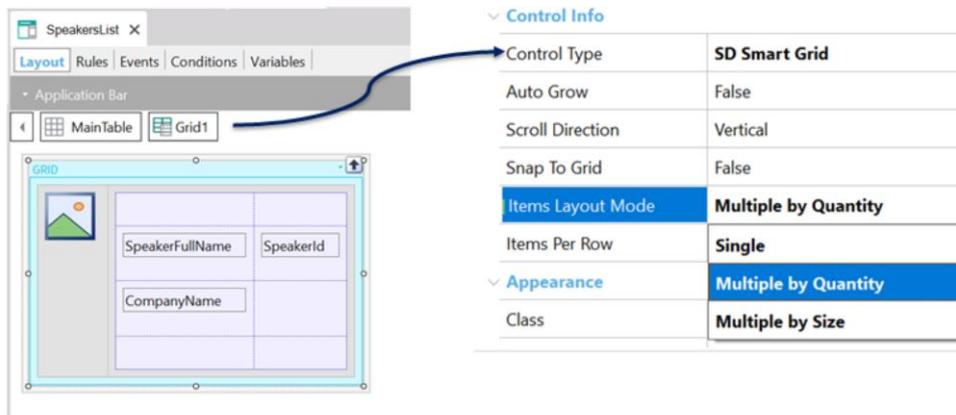
Vamos para a Seção General, no modo View e vamos modificar no atributo aqui também e usaremos o control type Rating, e deixaremos os valores por default.

Façamos o mesmo no modo Edit, alteramos o control type, vamos gravar tudo, executamos a aplicação novamente.

Vamos ver então no emulador, entremos em um Restaurante, agora já aparece o controle com as estrelas, editemos este restaurante, vamos marcar as que queremos, por exemplo 4 e confirmamos, e agora mostrará 4 estrelas, o mesmo que no List.

Vamos voltar para a apresentação.

## SD Smart Grids Control Type



Control Info	
Control Type	SD Smart Grid
Auto Grow	False
Scroll Direction	Vertical
Snap To Grid	False
Items Layout Mode	Multiple by Quantity
Items Per Row	Single
Appearance	Multiple by Quantity
Class	Multiple by Size

SD Smart Grids

Vejamos um tipo de controle que utilizamos na aplicação que vimos ao iniciar o curso quando mostrávamos, por exemplo, uma lista com os palestrantes, se trata de SD Smart Grid, que como o SD Maps, é um tipo de controle para Grids.

Quando o usamos no grid nos permitirá configurar certas propriedades, as mais importantes são:

- Auto Grow: indica se o tamanho será ajustado quando o conteúdo for maior que o container.
- Scroll Direction: indica se o scroll para mover de página é horizontal ou vertical.
- Items Layout Mode: tem 3 valores, Single, Multiply By Quantity e Multiply By Size, estes valores nos permitirão alterar o número de itens a serem exibidos.

## Demo: Using SD Smart Grid

Vamos ver como se utiliza no GeneXus.

Eu tenho preparado um Panel, neste caso uso o tipo de objeto Panel For Smart Devices, se chama SpeakersList e o adicionei ao Menu EventDay para que possamos visualizá-lo.

Este panel tem uma Main Table com uma altura fixa de 250 dips, para que possamos observar melhor o efeito do controle SD Smart Grid.

O grid é um grid padrão com os atributos SpakerImage, SpeakerFullName, SpakerId e CompanyName, e estou usando o Stencil SpeakerInfo que criamos anteriormente.

Nas propriedades do grid, vou mudar para usar o SD Smart Grid.

Por padrão, vemos Scroll Direction Vertical e a propriedade Items Layout Mode é Single, comportando-se assim como um grid padrão.

aparece somente um item porque items layout mode é single e o scroll é vertical.

Vamos alterar Items Layout Mode para usar Multiply By Quantity, com uma quantidade de 2.

Podemos ver que o grid conterà 2 itens por linha e a rolagem é vertical. coloquemos 3

itens, vão encolher cada um, quase não se vêem.

Observe que todas estas alterações podem ser feitas usando live editing diretamente. Vamos deixar 1 novamente.

Agora vamos alterar o scroll para que seja horizontal, essa é a configuração que usamos na aplicação que vimos no início do curso, com a opção Multiply by Quantity em 3 itens por coluna. Vamos tentar com 2, observe que mostra uma pequena parte da imagem à direita, indicando que há mais itens.

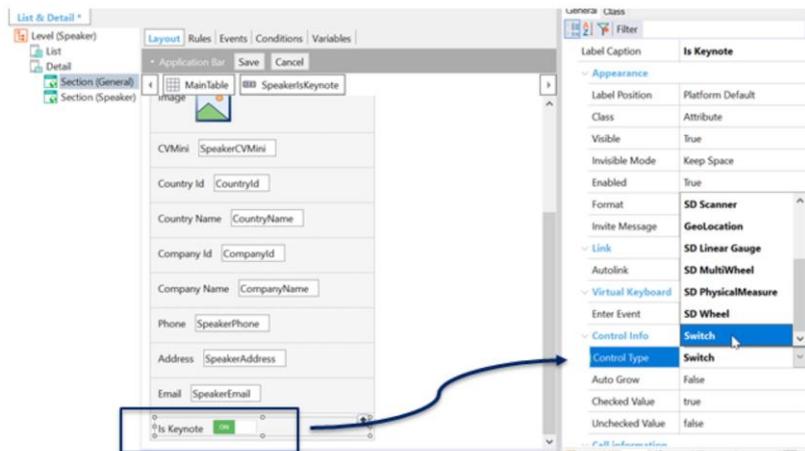
novamente, a visualização é atualizada automaticamente.

Vamos usar Multiply By Size agora, aqui temos que especificar uma altura mínima e máxima, tenha em mente que neste grid os itens têm uma altura de 80 dip, por isso tem esse valor mínimo. Vamos colocar 100 na altura mínima, serão mostrados apenas 2 itens, porque temos 200 dips das linhas e apenas 250 da altura da tabela, se se lembram, o que excede o máximo e mostra apenas 2 itens. e o que acontece se colocarmos 200? Apenas um item será visto. Perfeito..

Voltemos para a apresentação para ver mais um tipo de controle.

## Switch Control Type

Switch



Outro tipo de controle muito utilizado é o controle Switch, é usado para apresentar atributos ou variáveis com valor Booleano.

Por exemplo, na Transação Speaker, temos o atributo SpeakerIsKeynote, que indica que o Palestrante é um palestrante destacado.

## Demo: Using Switch Control Type

Vamos fazer isso diretamente em nossa aplicação.

Entramos no `WorkWithDevicesSpeaker` e vamos para a seção `General`, estamos no `Modo Edit`, este atributo não queremos mostrá-lo no modo `View`, pois é um atributo que só nos interessa no `BackEnd`.

Simplemente mudamos o `control type default` por `Switch`, gravamos e rodamos a aplicação.

Entremos em `Speakers`, entramos em um `Palestrante` e vamos `Editá-lo`.

Bem, lá já vemos o controle, podemos ligá-lo ou desligá-lo, isso irá gravar um valor `True` ou `False`, o deixamos em `True`. Perfeito.

## More Control Types

### For Attributes or Variables

-  SD Scanner
-  SD Linear Gauge
-  SD Multi Wheel
-  SD Physical Measures
-  Chronometer

### For Grids

-  SD Charts
-  SD Image Gallery
-  SD Paged Grid
-  SD Matrix Grid

Existem muitos mais controles que podemos usar em nossas aplicações.

Por exemplo para Atributos ou Variáveis, podemos usar SD Scanner para ler dados de um código de barras ou um código QR, SD Linear Gauge para exibir informações em intervalos lineares ou circulares, SD Multi Wheel que mostra uma roda para estabelecer os valores, SD Physical Measures que é muito útil quando trabalhamos com medidas físicas como temperaturas, medidas de peso ou volume e um Cronômetro, o que nos permitirá executar eventos automaticamente quando, por exemplo, um certo tempo for alcançado.

Para os Grids também temos mais tipos de controles, por exemplo, SD Charts para apresentar gráficos de pizza ou timelines. SD Image Gallery para mostrar imagens, SD Paged Grid que tem um modo especial de pagnar e mostrar o número de páginas dos grids ou SD Matrix Grid que nos permite trabalhar com informação complexa em tabelas bidimensionais.

Esta é apenas uma lista limitada, convido vocês a buscar mais informações sobre estes controles e mais no wiki do GeneXus.

<https://wiki.genexus.com/commwiki/servlet/wiki?9550>

# GeneXus™

Videos	<a href="http://training.genexus.com">training.genexus.com</a>
Documentation	<a href="http://wiki.genexus.com">wiki.genexus.com</a>
Certifications	<a href="http://training.genexus.com/certifications">training.genexus.com/certifications</a>