

# Menus e Master Pages

Telas interativas

*GeneXus 16*

## Action groups

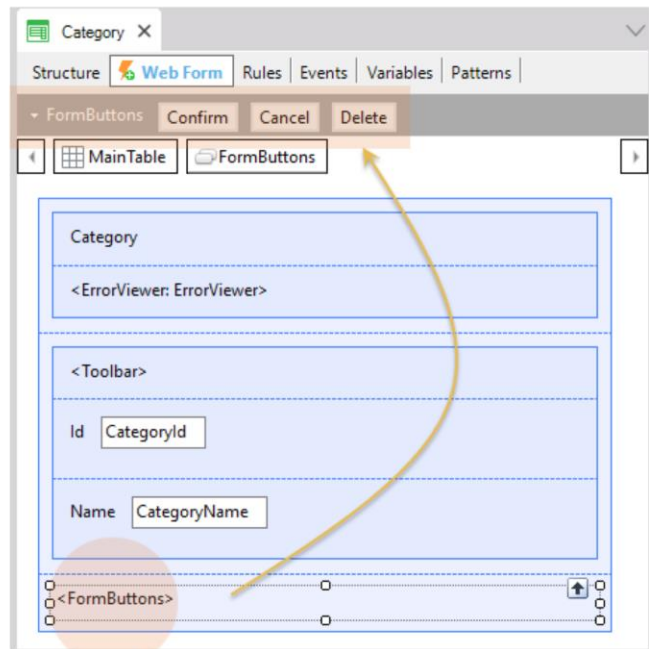
The screenshot shows a web browser window with the URL <https://trialapps3.genexus.com/id6489467c774700f9376addef14ad2c...>. The page title is "Application Name" and it features the "GeneXus" logo. The main content area is titled "Category" and contains a form with the following elements:

- Navigation buttons: <<, <, >, >>, and SELECT.
- Input field for "Id" with the value "13".
- Input field for "Name" with the value "Museum".
- Action buttons: CONFIRM, CANCEL, and DELETE.

Se abrimos uma transação qualquer em execução, por exemplo Category, vemos na tela os botões de navegação antes dos atributos, e abaixo, os botões para confirmar, cancelar ou eliminar a categoria com a qual estamos trabalhando em cada oportunidade. Onde se configura isto no objeto transação?

## Action groups

- Action group: elemento da UI utilizado para agrupar controles (botões, imagens, text blocks e variáveis)
- Ajuda a criar facilmente uma toolbar
- Permite inserir vários controles dentro de uma mesma célula de uma Responsive Table. Será exibida em uma única linha

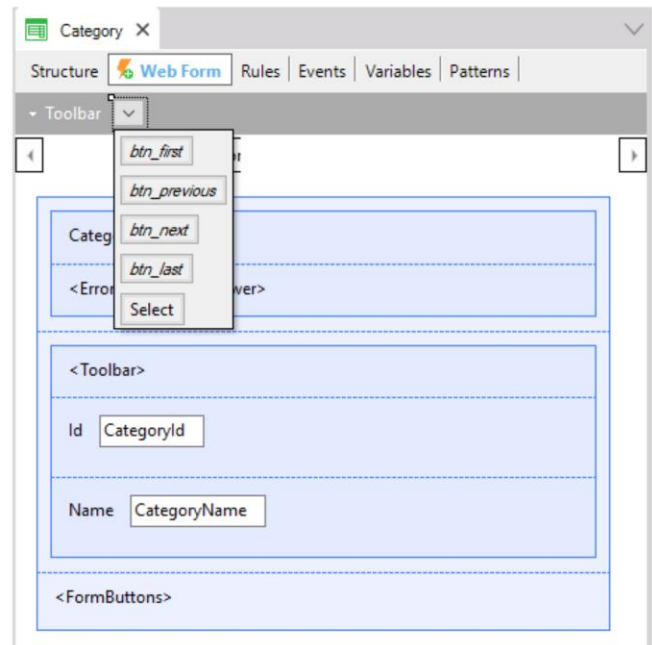


Se abrimos o Web form, vemos que, além dos atributos que compõem a estrutura da transação, GeneXus incorpora dois controles conhecidos como Action Groups.

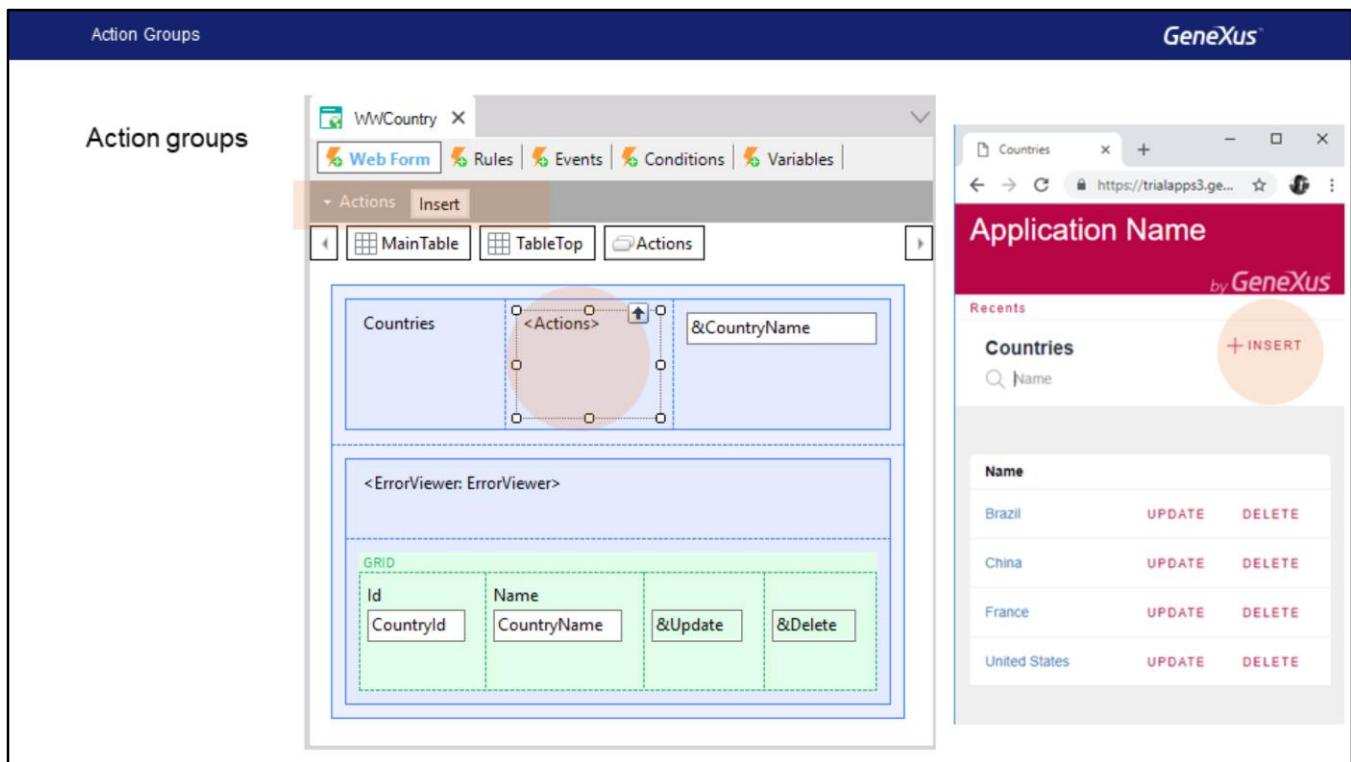
Na imagem estamos editando o controle <FormButtons>, e vemos que ao fazê-lo aparecerá acima do Web form, na seção conhecida como Action Bar, de quais controles será composto este grupo de ações. Vemos que corresponderá aos botões para confirmar, cancelar ou excluir a categoria.

## Action groups

- Action group pode conter, entre outros controles, outro controle Action group.



Se agora nos posicionamos sobre o action group control <Toolbar> no form, vemos que se edita acima, na Action Bar, o grupo de ações chamado Toolbar. E vemos que tem um sub-action group, ou seja, um action group dentro do outro action group. Este action group contém os botões para navegar entre os diferentes registros da transação e um botão Select, que como sabemos, chamará a lista de seleção de categorias, criada automaticamente pelo GeneXus.



Também no web panel Work With, implementado pelo GeneXus ao aplicar o pattern, aparecerá um action group, que em execução corresponde ao botão Insert.

## Action groups

The screenshot displays the GeneXus Action Groups interface. At the top, there is a search bar labeled "Attractions" with a search icon and a "Name" placeholder. Below the search bar is a table of attractions. The table has columns for Id, Name, Country Name, Category Name, Photo, City Name, and Trips. The data rows are:

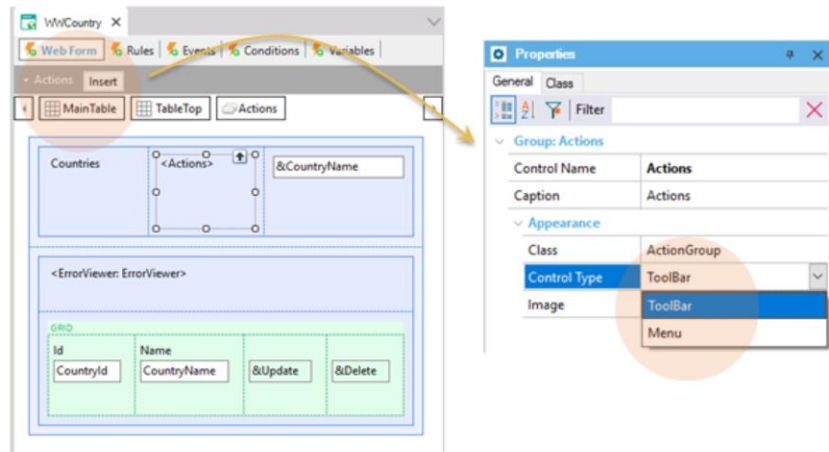
Id	Name	Country Name	Category Name	Photo	City Name	Trips
25	Christ the Redemmer	Brazil	Monument		Rio de Janeiro	3
24	Eiffel Tower	France				
28	Forbidden city	China				

Below the table, there is a configuration window for the "Attractions PDF" action group. The window has a title bar "WwAttraction X" and a menu bar with "Web Form", "Rules", "Events", "Conditions", and "Variables". The "Actions" tab is selected, and the "Attractions PDF" action is highlighted. The configuration area shows a "MainTable" and "TableTop" section. The "Attractions" table is visible, with a "Hide Filters" button and a "Name" filter. The "Ordered By" section shows "Name" and "Country". The "ErrorViewer" section shows a "GRID" with columns: Id, Name, Country Name, Category Name, Photo, City Name, Trips, &Update, and &Delete. The "Attractions PDF" action is configured to use the "&AttractionName" variable.

Quando, na instância do pattern Work With aplicado à transação Attraction adicionamos uma ação para imprimir as atrações em pdf, vemos como GeneXus incorporou no objeto web panel gerado, essa ação ao action group correspondente.

## Action groups

- Podemos criar nossos próprios action groups em web panels e transações e adicioná-los no form que desejemos.
- Dois tipos de action groups:
  - ToolBar
  - Menu

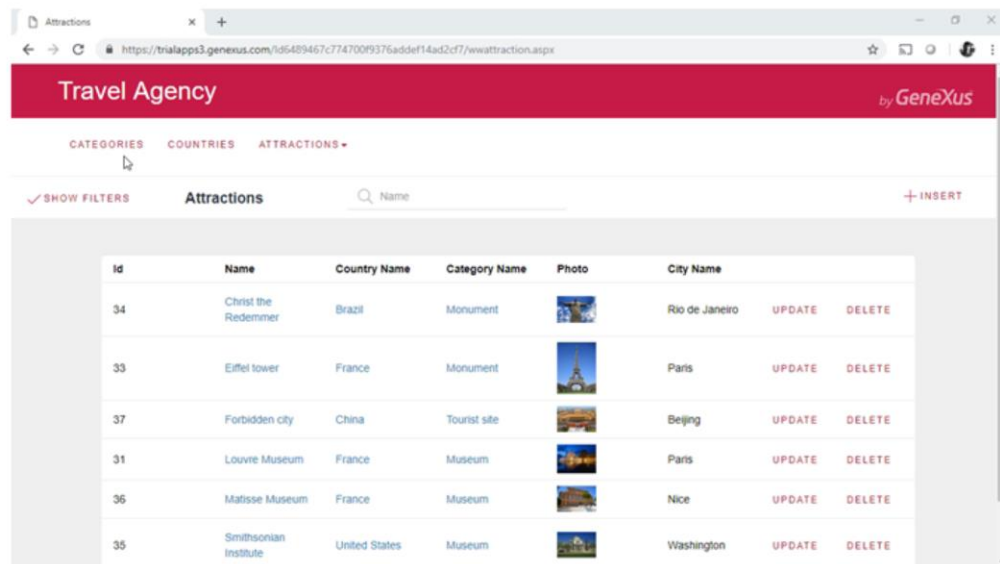


Os exemplos de action groups vistos antes são do tipo ToolBar. O vemos ao nos posicionar na Action Bar, em suas propriedades.

Em seguida, implementaremos um menu de ações.

## Menu de navegação para o backend

- Backend vs Frontend
- Queremos um menu de navegação com as diferentes opções (aqui reduzidas) para o backend.



[ DEMO: [https://youtu.be/yiM-ss0\\_NSU](https://youtu.be/yiM-ss0_NSU) ]

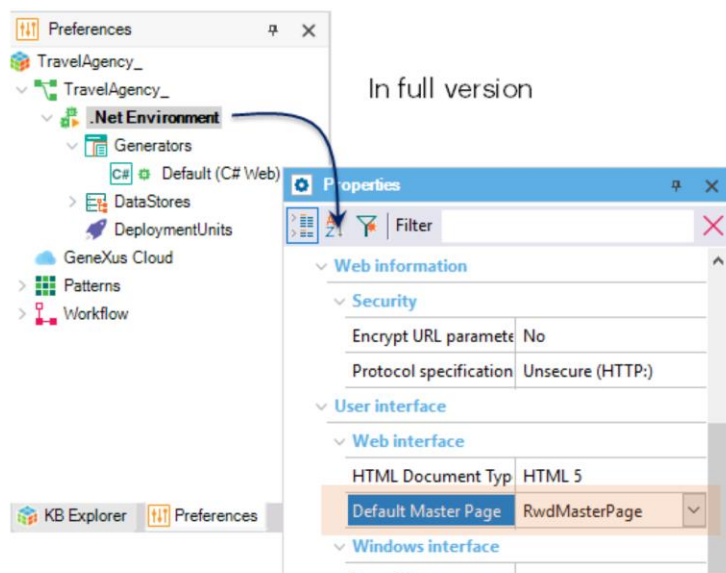
Vamos querer adicionar um menu que aparece horizontal quando o tamanho da tela permite, e como o típico menu de hambúrguer (deslizante) quando não (por exemplo, quando a app web é executada em telefones). Na primeira demonstração, vemos o que queremos alcançar. É uma versão simplificada, na qual não colocamos todos os itens que nossa app precisa (por exemplo, não adicionamos os painéis que desenvolvemos aulas atrás).

As apps em geral têm o que se conhece como **frontend**, ou seja, a aplicação que os clientes utilizarão, a app em si, e um **backend**, que também é uma app web, mas que será utilizada pelos responsáveis por manter os dados da aplicação, ou seja, é uma aplicação interna da empresa. Temos trabalhado até agora de forma não integrada. Não temos prestado muita atenção a estes aspectos. Começaremos a integrá-los um pouco no que segue. A maior parte do que fizemos será para o backend. Por exemplo, os trabalhar com, serão utilizados pelo pessoal da Travel Agency para carregar dados, e não pelos clientes que desejam procurar viagens, países para visitar, etc. Inclusive os web panels e relatórios que implementamos parecem mais apropriados para o backend. Não há nenhuma diferença funcional entre backend e frontend, exceto por um tema de segurança e permissões. Nós somos aqueles que determinam o que se implementa no backend e o que vai para o frontend.



## Menu de navegação para o backend

- Aplicamos o Work With for Web a todas as transações, para o nosso backend.
- Vamos invocar cada WW a partir de um item do menu
- Queremos que o menu seja visível em todas as páginas web de nossa app
- Onde o inserimos? → Objeto MASTER PAGE
- Como o cada objeto sabe com qual Master Page executar?



O que faremos, então, será adicionar um menu que seja visto em todas as páginas do backend. Temos testado com o Developer Menu mas, obviamente, essa não pode ser uma página de entrada (home) da aplicação. Começaremos por aplicar o pattern Work With às transações que não o tenham aplicado.

O menu, como dissemos, será um tipo especial de action group. Em qual objeto web o inserimos para que apareça como menu em todas as páginas que visitemos dentro do backend?

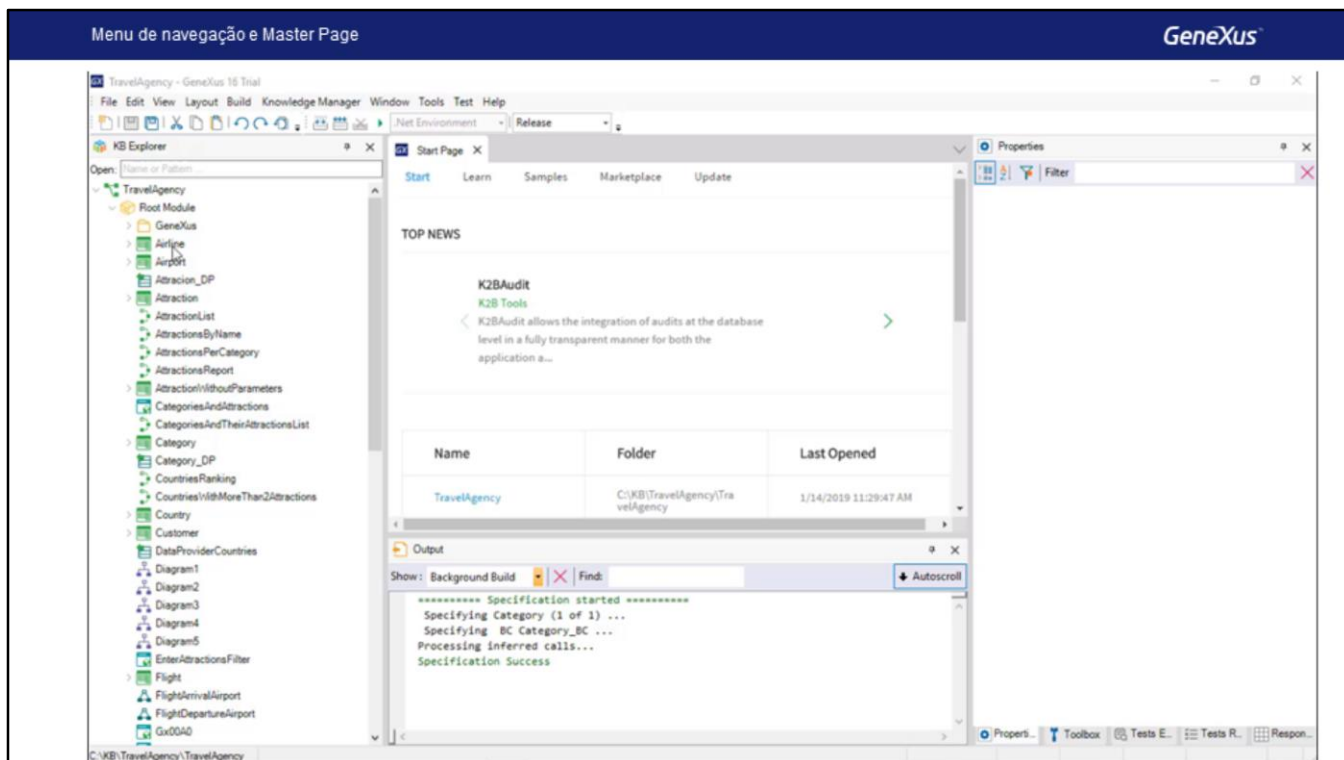
Já tínhamos falado das páginas mestre. Dissemos que toda a KB se criava com estas páginas, que são o quadro dentro do qual são carregadas todas as páginas executadas.

Por padrão, cada Environment (mais adiante falaremos sobre os environments, no momento pensemos que é a instância da app que estamos desenvolvendo) tem definida uma Master Page associada a todos os seus objetos web (incluindo aqueles que criamos do zero).

Na demonstração a seguir veremos, por exemplo, que a transação Attraction é do tipo Web Page e vemos que tem como Master Page a de nome RwdMasterPage. Outras Master Pages predefinidas vêm com a KB, para outras coisas que não veremos aqui.

Então, se olharmos para o objeto WWAttraction criado pelo pattern, também vemos que tem definida essa Master Page como sua página mestre. O mesmo vemos se visualizamos as propriedades de um web panel que nós criamos. Onde se encontra esta Master Page? Dentro da pasta GeneXus, subpasta Web

Se estivéssemos usando a versão full do GeneXus em vez da Trial, poderíamos acessar a aba Preferences, onde poderíamos alterar essa Master Page default de todas as nossas páginas web, por outra. De qualquer forma, sempre pode, para um objeto específico, alterar a Master Page por qualquer outra definida na KB. Na versão Trial não podemos modificar a Master Page default. Será sempre a que se vê na imagem acima.



[ DEMO: <https://youtu.be/eCuD6QQsnyU> ]

Na demonstração estamos vendo como absolutamente todas as páginas, tanto as do work with, como as das transações e inclusive a dos web panels que nós mesmos criamos estão sendo executados dentro dessa estrutura, onde, como podemos ver, toda a barra de cima permanece inalterada. Queremos que o menu esteja nessa área.

Abrimos a master page RwdMasterPage, e modificamos, primeiro o nome da app, para que agora diga Travel Agency. Poderíamos colocar uma imagem lá, como um ícone da app, e tudo o que queremos.

Também vemos que aparece um controle de tipo Component, já mencionado no curso. Esse componente é criado invocando outro objeto web panel que é responsável por exibir os objetos visitados. É o que vimos em execução como Recents.

E então, o mais importante, é o controle <ContentPlaceHolder>. Este controle só pode estar dentro de um **web panel do tipo Master Page**. É onde será carregada cada página invocada. O restante é o quadro que essa página terá.

Eliminamos o <component> porque não queremos a rota dos links mais visitados e, em seu lugar é onde colocaremos nosso menu.

Para fazer isso, vamos para a Toolbox e arrastamos o controle New Action Group para o seu lugar. Uma pequena janela se abre para nomear o action group que estamos criando. O chamamos

NavMenu. Poderia ser um menu dinâmico, mas não o veremos aqui.

Onde definimos os itens do menu? É acima, na Action Bar que é mostrada em cinza no web form. Clicamos com o botão direito e inserimos o que queremos, por exemplo, um Text Block. Modificamos suas propriedades alterando o nome para o controle, porque então teremos que nomeá-lo nos eventos, e o Caption, porque será o que queremos que diga em execução.

Fazemos o mesmo para cada opção que desejamos incluir.

Então vamos aos eventos, e como uma primeira coisa comentamos ou eliminamos dentro do evento Refresh a carga do componente que eliminamos. Caso contrário, nos mostrará um erro ao querer gravar, porque esse componente já não existe mais no form.

E então no evento Start o que fazemos é definir os links que queremos para cada uma das nossas opções de menu. Isto é feito por meio da propriedade Link do controle TextBlock e, em seguida, invocando o objeto desejado com o método link. Aqui não vamos entrar nas diferenças entre invocar com ou sem Link.

Pressionando F5 e abrindo qualquer objeto web, vemos que o menu aparece. Aqui, incluímos apenas como itens as atrações, categorias e países, para simplificar. Mas isto não é um menu de navegação, é uma ToolBar. Não está aparecendo como queríamos. Não mostra o menu de hambúrguer quando o tamanho da tela é pequeno. É que não mudamos o seu Control Type para Menu.

Fazendo isso e pressionando F5, o vemos como queríamos. De fato, se usando chrome pressionarmos F12, podemos fazer variar a tela para ver como ficaria não apenas nos desktops, mas também nos dispositivos móveis. É uma boa maneira de testar a responsividade. E assim vemos como em um iphone, por exemplo, está aparecendo o menu de hambúrguer.

## Vimos

- Propiedade **Master Page** de cada objeto web
- Objeto Master Page
  - Controle ContentPlaceholder em seu web form
  - Como inserir um action group
  - Como fazer que seja do tipo menu



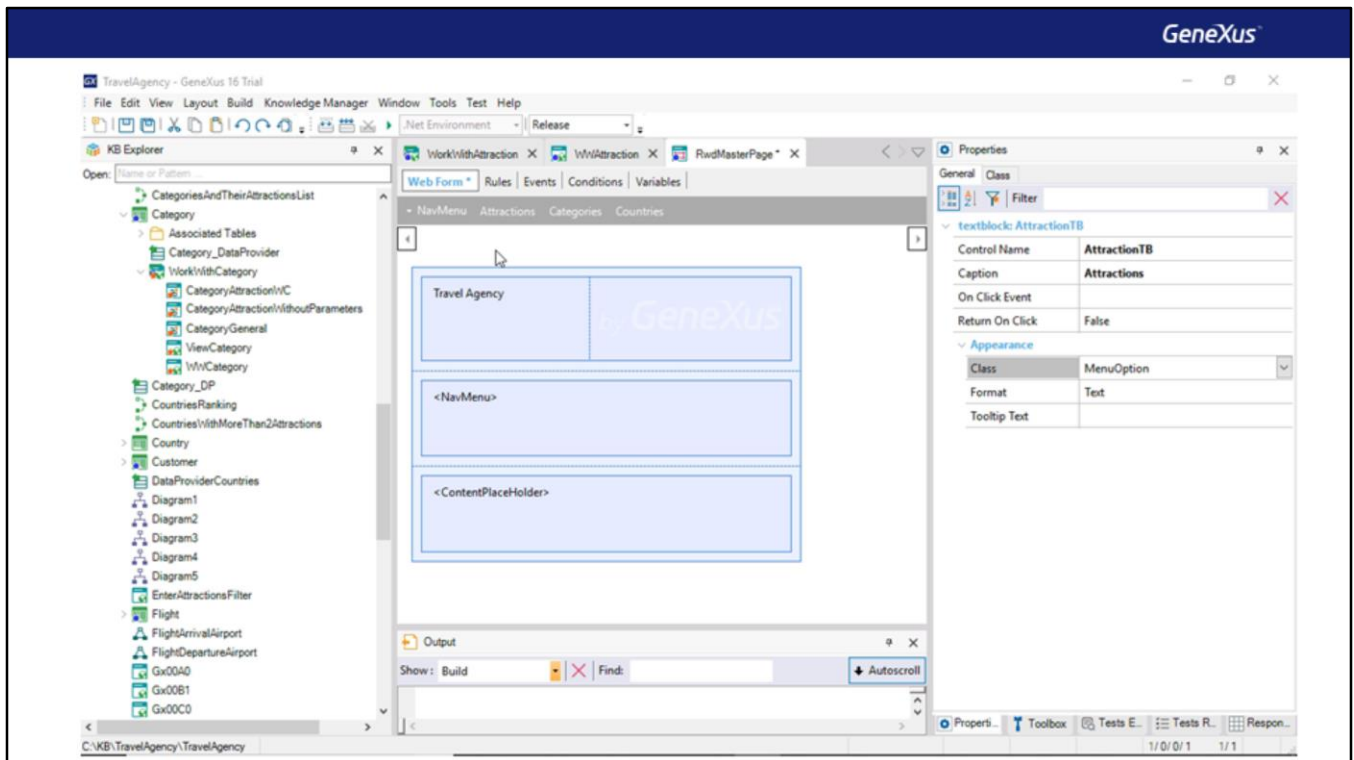
DEMO

## Menu com sub action group

- É possível aninhar action groups dentro de outros
- Para alterar o design ou algo do comportamento dos controles, os associamos a classes, que ficam definidas dentro de um objeto Theme da KB (aqui apenas o apresentamos; será visto especificamente mais adiante)



Veja a demonstração na próxima página.



[ DEMO: <https://youtu.be/zPFotMPpw6Q> ]

Para adicionar um action group a outro, basta clicar com o botão direito do mouse / Insert Group. Criaremos um submenu para mostrar coisas das atrações turísticas: tanto o Work With como alguns dos relatórios que implementamos em nossa KB. Também poderiam ser web panels.

Ali dentro nós arrastamos o text block que tínhamos no pai, Attractions, e adicionamos dois novos text blocks, para invocar cada relatório. Temos que colocar um Caption apropriado para o Submenu, porque será o que se verá antes de abri-lo. Colocamos como Caption, Attractions.

'Vemos em execução que quando o menu é exibido horizontal -não como hambúrguer- o submenu está sobreposto ao conteúdo abaixo. Ou seja, sua cor de fundo (background color) é transparente. Se quiséssemos que fosse branco, para que isto não aconteça, onde o modificamos?

Se no GeneXus editamos as propriedades do action group que implementa o menu (o mesmo veríamos se editamos as do submenu), descobrimos que sob o grupo Appearance aparece uma propriedade class, por padrão a de nome ActionGroup. O que é uma classe? Já dissemos quando vimos mais sobre web panels, que a relação entre um controle do form e sua classe é análoga à relação entre um tipo de dados de um controle atributo ou variável e seu domínio. Ou seja, a classe centraliza muitas propriedades, de forma a ser possível alterá-las em apenas um local e aplicá-las automaticamente a todos os controles que a tenham por classe. Vemos que GeneXus nos oferece um caminho curto para poder editar as propriedades da classe do controle que estamos editando. É a tab Class, ao lado de General. Ali, alteramos a propriedade

Background Color para White.

Mas ao fazer isso, vemos que é automaticamente aberto para nós um objeto chamado Carmine. É um objeto do tipo Theme e é o theme predefinido da KB. É que, ao alterar para White o valor anterior, não o fizemos apenas para o controle action group que estávamos editando, mas para qualquer outro action group que tivesse por classe a de nome ActionGroup. O objeto theme é aquele que armazena as propriedades de todas as classes possíveis da KB.

Vemos que há também a classe ActionGroupItem, que é a que permite definir o design e comportamento dos itens do menu. Assim, se observamos, vemos que a cor avermelhada que vimos em execução nas opções do menu vem da propriedade Forecolor desta classe.



# GeneXus™

**The power of doing.**

Videos

[training.genexus.com](http://training.genexus.com)

Documentation

[wiki.genexus.com](http://wiki.genexus.com)

Certifications

[training.genexus.com/certifications](http://training.genexus.com/certifications)