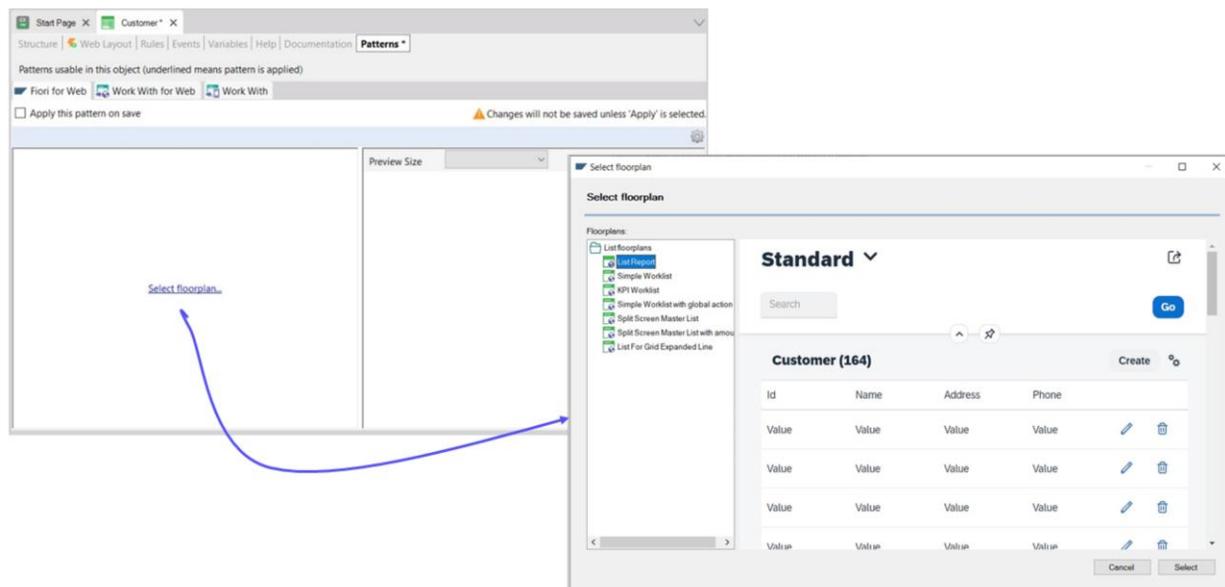




Using patterns: Fiori for Web

Part 1

Fiori for Web



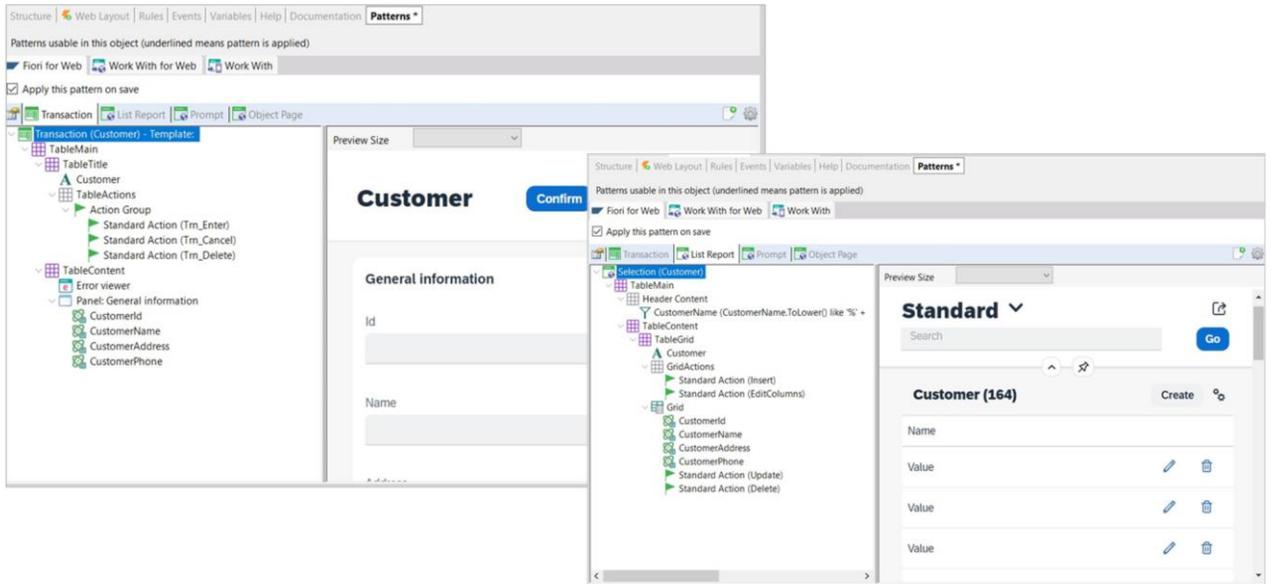
Once the Fiori for Web pattern has been initialized with the corresponding design system, we will apply it to the Customer transaction.

In this option, we can select the floorplan that we want to use for this pattern in this transaction.

We click on it... And see that it opens this screen. Let's look at the first one: List Report. It shows a preview of the information and functionalities provided. Here we see that it will show a grid with the identifier and name of each customer, with the possibility to update or delete them. It is also possible to insert a new one, in addition to filtering what is seen in the grid.

The other floorplans are different ways, with more or less functionality, of working with the Customer transaction data; i.e. with the list of customers.

Fiori for Web: ListReport



We choose, then, the List Report floorplan... and these tabs appear, each one showing the objects or screens that will be associated with it, as we will see in a moment at runtime.

For example, here is a screen similar to that of the transaction. It will be used when we want to insert, modify, or delete a customer instead of using the web form of the transaction. And here we will see the List of customers.

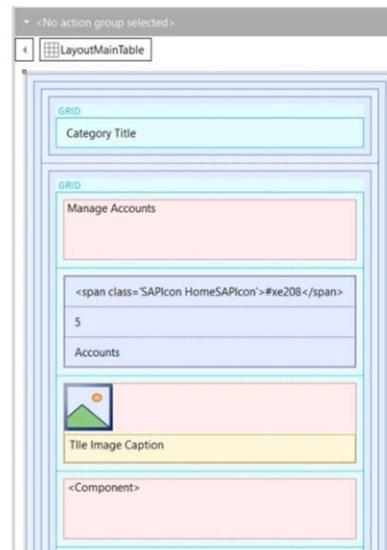
In addition, we can see that this check box is selected (Apply this pattern on save):

We save, and note what happens in the KB Explorer window with the Customer node.

All the objects that GeneXus will create to implement the screens and functionalities of this Fiori pattern for this transaction will appear under the transaction.

Fiori for Web: ListReport

Environment: JavaSQLServer	
User Interface	Web
Language	Java
Reorganization Generator	Default
Data Source	SQL Server
Target Path	JavaSQLServer002
Translation type	No translation
Startup Object	FioriLaunchpad



Before pressing F5, let's go to the Preferences. Here we have a single Environment for this knowledge base, although we could have many.

In its properties, we find one named Startup Object, which indicates what will be the application's main object; that is, its entry point. Before applying the pattern, it was set to "none" and that is why the GX launchpad was displayed when pressing F5.

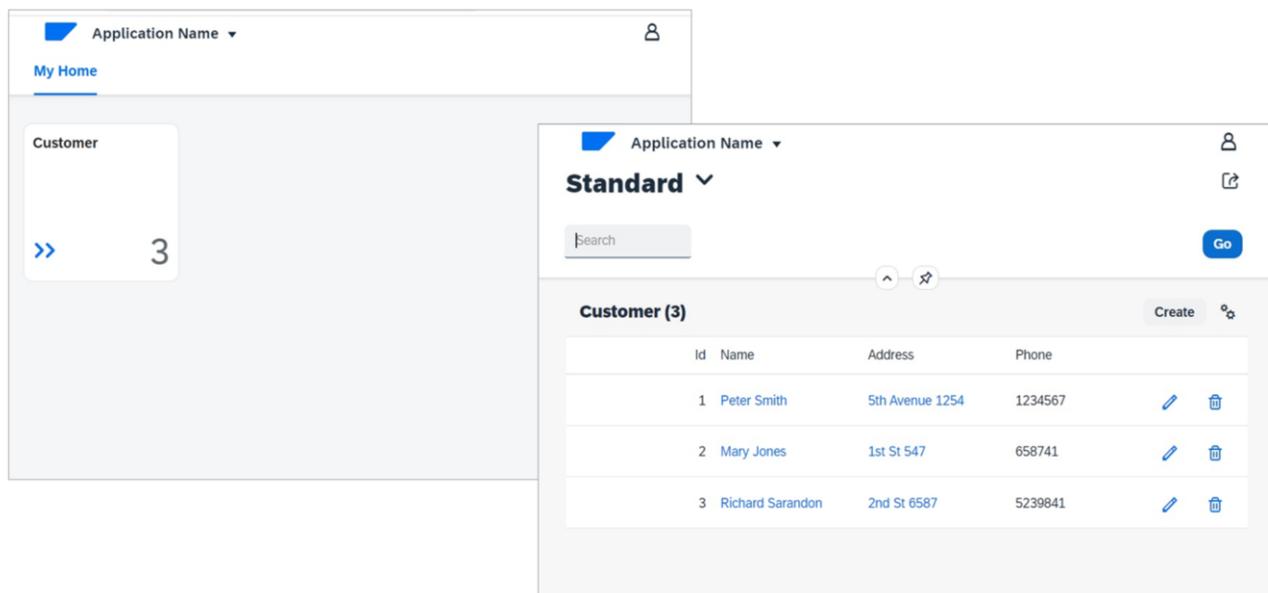
After applying the pattern, we see that the startup object is called FioriLaunchpad.

Where is this GeneXus object that will now be the object compiled when pressing F5, along with its entire invocation tree?

It will be in the FioriBaseObjects module that was imported when we pressed Initialize Fiori a while ago.

It is a GeneXus object of Web Panel type, i.e. it implements a web screen.

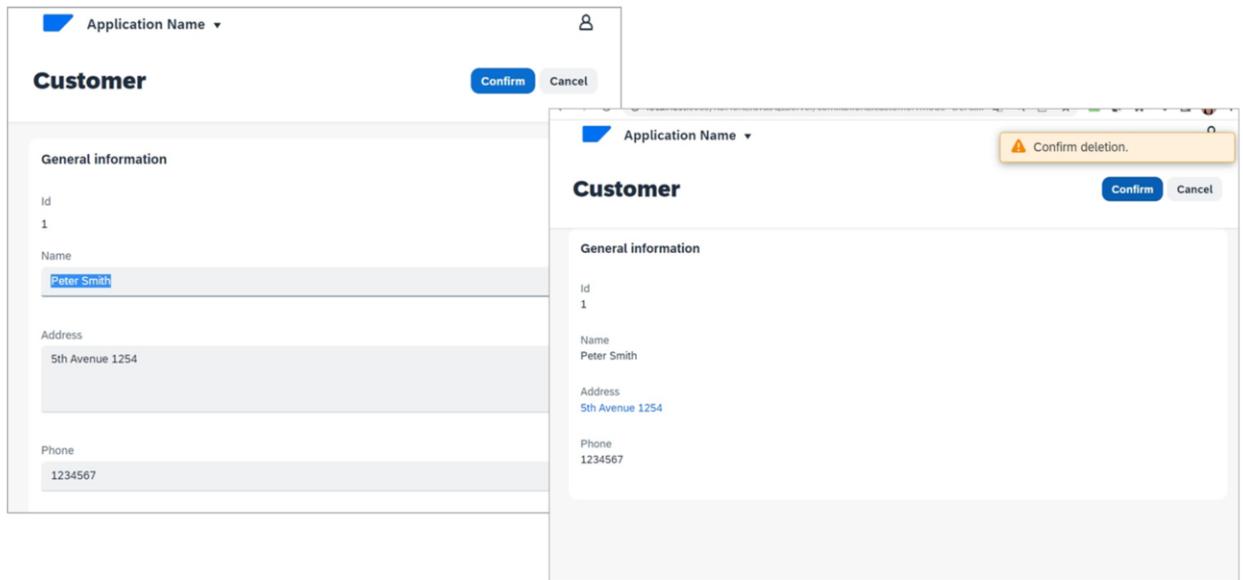
Fiori for Web: ListReport



Now, let's press F5 to run everything that has been automatically generated by applying the pattern.

The Launchpad with Fiori design is displayed in the browser, offering a card with a link to work with customers. It is also indicating the number of registered customers:

Fiori for Web: ListReport



We can perform operations such as displaying the complete information of a given customer.

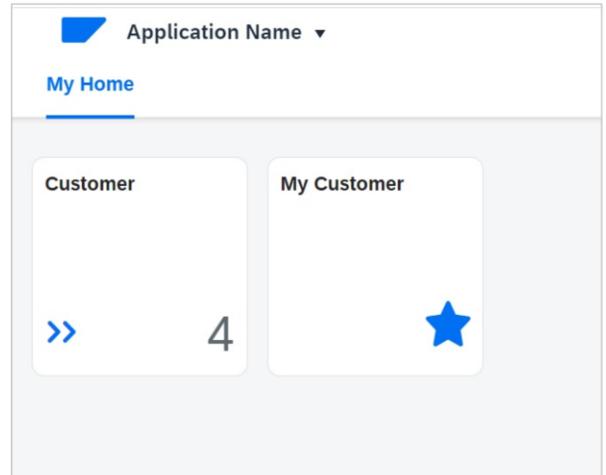
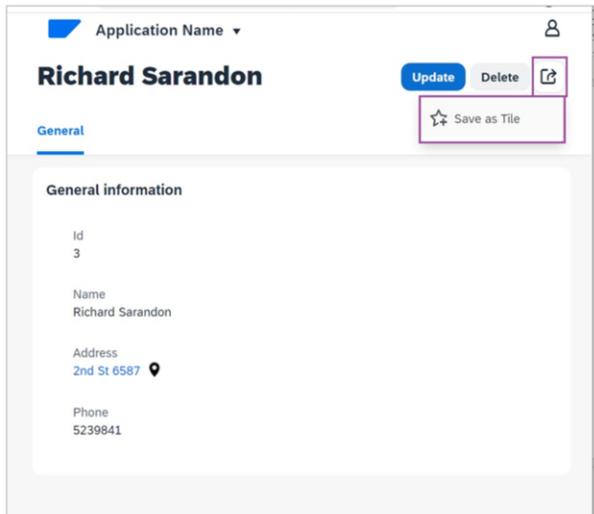
There we have the General tab that shows the customer's information and buttons to Update that information or Delete the customer. Also, there is a SalesOrder tab that will show all the sales orders recorded for the customer being displayed:

We can also update their information directly.

This screen is similar to that of the Customer transaction, where the navigation buttons have disappeared because we are positioned on the chosen customer.

In the same way, we can delete the customer. We see that since it is calling a screen that has the transaction logic, in DELETE mode, if the customer has related sales orders then it will not allow us to delete it.

Fiori for Web: ListReport



Let's add a new customer: We can perform filters by name on the displayed customers. And save a specific filter:

We can also save a specific List Report option as a card (tile) that will be displayed in the launchpad:

And from here we return to the launchpad:

Fiori for Web: ListReport

The screenshot shows a Fiori List Report for 'Customer' in the 'Standard' application. The report displays a table with 4 records. A 'Select Columns' dialog box is open, allowing the user to customize the columns shown in the table. The dialog box has a title bar with 'Application Name' and 'Standard *'. It contains a search field and a 'Go' button. The table has columns for 'Id', 'Name', 'Address', and 'Phone'. The 'Select Columns' dialog box has a title 'Select Columns' and a close button 'X'. It contains a list of columns with checkboxes: 'Id' (unchecked), 'Name' (checked), 'Address' (checked), and 'Phone' (unchecked). At the bottom of the dialog box are 'OK', 'Cancel', and 'Restore' buttons. The table data is as follows:

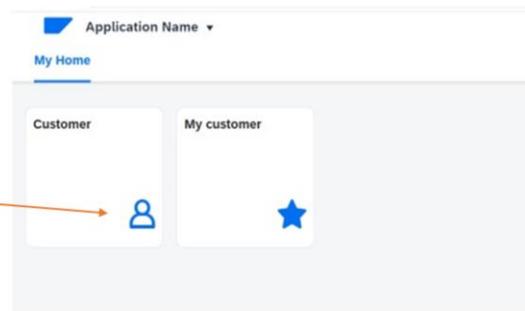
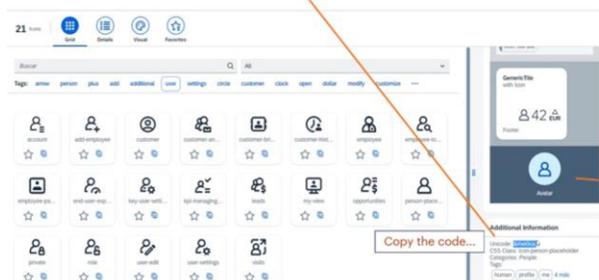
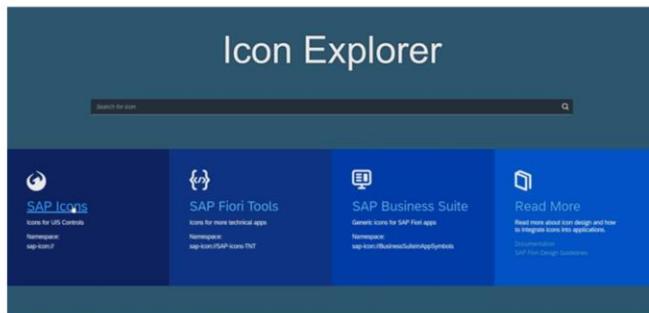
Id	Name	Address	Phone
2	Mary Jones	1st St 547	658741
3	Richard Sarandon	2nd St 6587	5239841
4	Ann Jones	4th Avenue 325	125478
5	Annabelle Parker	5th Av 5478	6588888

We can also customize the grid data we want to see at runtime. For example, only name and address.

Fiori for Web: ListReport

Launchpad

Include In Launchpad	True
Tile Category	
Tile Type	Information
Tile Icon Unicode	
Tile Indicator	



OK. What is displayed and the behavior of all these screens can be customized. For example, suppose that in the card corresponding to customers we do not want to see the number of customers indicator, and we want to see an icon that represents them.

If we go to the List Report tab, at the root, we see these properties.

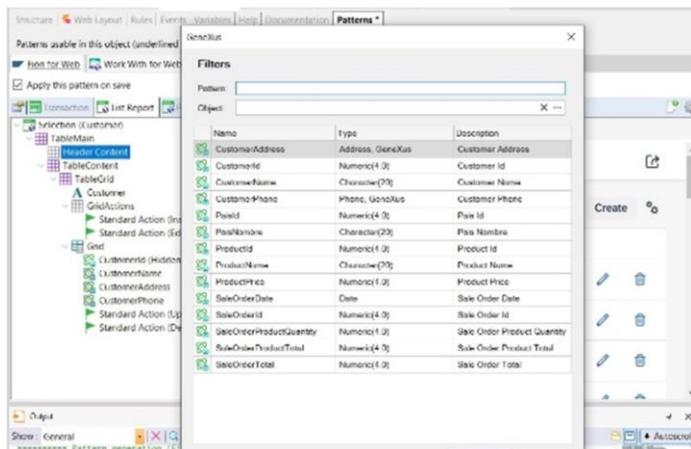
This property indicates whether a tile will be included in the launchpad with this List Report. If we change the value to False, it will no longer be displayed. This other property allows us to configure a specific icon, or rather, the icon code according to the Fiori design guidelines:

This property specifies whether an indicator will be included in the card. The <default> value corresponds to a count of the grid's base table, which in this case is the Customer table, but we could change it to a formula or not indicate anything, and in the latter case no indicator will be displayed on the card.

We press F5:

Fiori for Web: ListReport

Responsive Customization	
Visible (Extra Small)	False
Visible (Small)	True
Visible (Medium)	True
Visible (Large)	True



At runtime, it looks like this:. OK. Let's now customize the instance of this pattern applied to the Customer transaction.

For example, let's suppose that we do not want to display the Customer's identifier.

For that, in the List Report tab, we see a Grid node with all the attributes that are being displayed in the form when the screen size allows it, in addition to the Delete and Update actions seen in the running grid.

We eliminate those that we never want to see, or set the Visible property to False. Note also that under Responsive Customization we configure visibility according to screen sizes

We have seen that by default the user is offered to filter the customers they want to see in the grid by their name. But where is this filter configured so that we can customize it, delete it, or create a new one?

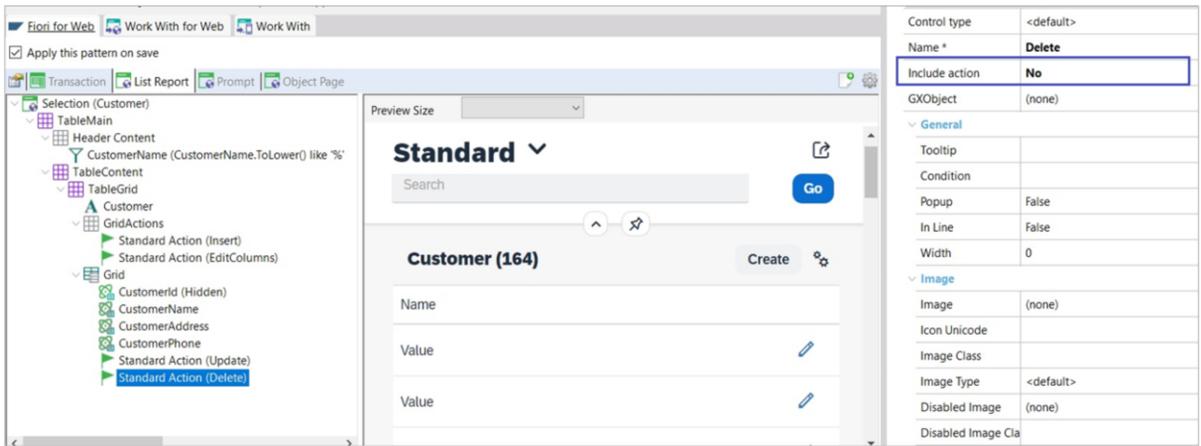
Below the HeaderComponent node:

If we delete it, we see how it automatically disappears from the preview:

Let's try reinserting it.

We right-click, select Add Filter attribute... and choose CustomerName.

Fiori for Web: ListReport



We may also want to disable some or all of the Insert, Update, and Delete actions for this panel.

To do so, we go to the GridActions node, where we have the two actions that we saw running: one to insert a new client, and the other to edit the columns displayed.

For example, if we don't want to offer the possibility to create a new customer, we set this Include action property to No, and see that it disappears from the preview.

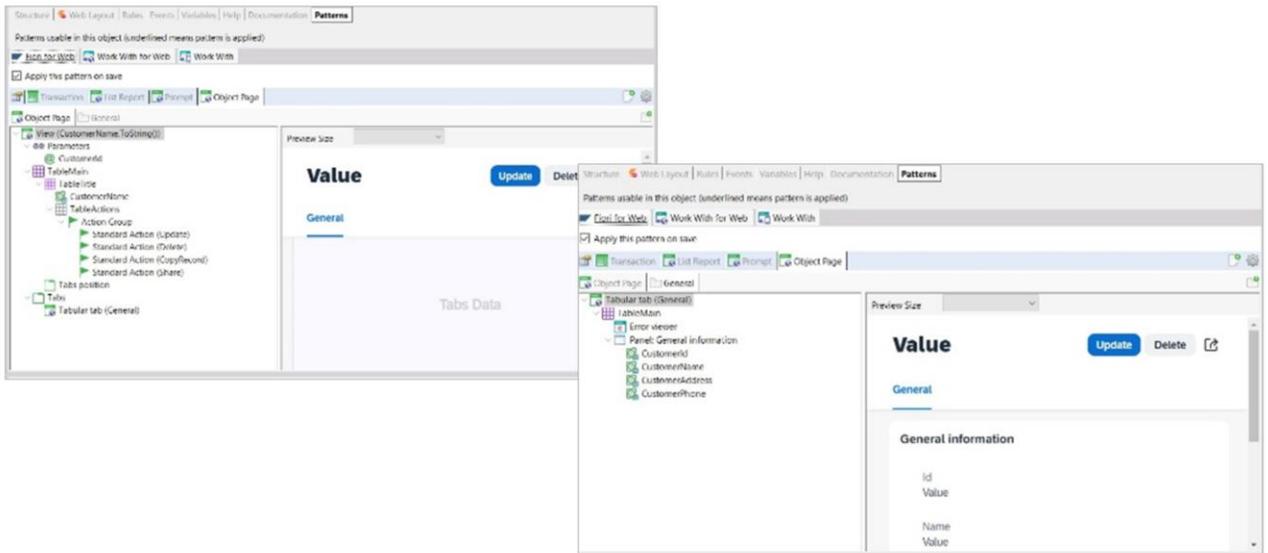
If we set it back to <default>, it reappears.

The Delete and Update actions are inside the Grid node, since they are shown for each row of the grid. So here we modify the <default> value for Delete and set it to NO.

In the preview, we can see how the delete option has disappeared. Let's press F5 to see it running.

:

Fiori for Web: ListReport



Good. So far, we have only seen some actions for the Customer List Report panel, but we can also customize the panel that shows the information of the selected customer.. The definitions of this panel are those found in the Object page tab:

In the General tab we try not to display the identifier, so we set its Visible property to False, and it is removed from the preview:

In the same way, we could customize the SalesOrder tab.

In this way, we have applied the ListReport floorplan to the Customer transaction, and seen a sample of what we can achieve with very little effort.

GeneXus[™]
by **Globant**

training.genexus.com