

Work With Pattern for Mobile



Diego Marranghello

Work With Pattern for Mobile



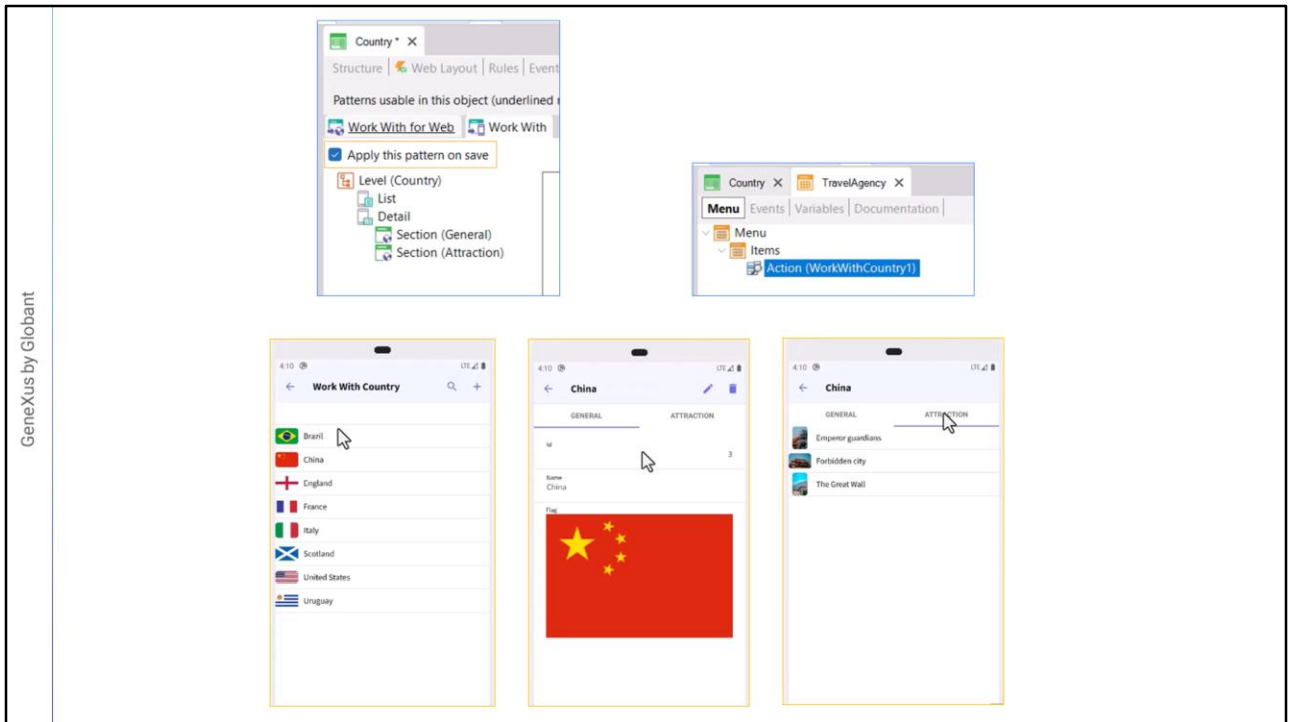
In this video, we will see the usage and features of the WorkWith pattern for mobile devices.

The use of this pattern is similar to the one for web.

Applying this pattern allows listing and handling database records in a friendly, attractive, and efficient interface.

The WorkWith pattern automatically generates screens to perform basic CRUD operations: Create, view, update and delete records.

In addition, it allows you to sort and filter the displayed data.



Let's see how to do it in GeneXus.

We have already created some transactions and other objects of an application for a travel agency.

Let's apply this pattern to the Country transaction.

We select the transaction, go to the Patterns section, and apply the WorkWith pattern.

After saving, the WorkWithCountry object is generated inside the transaction.

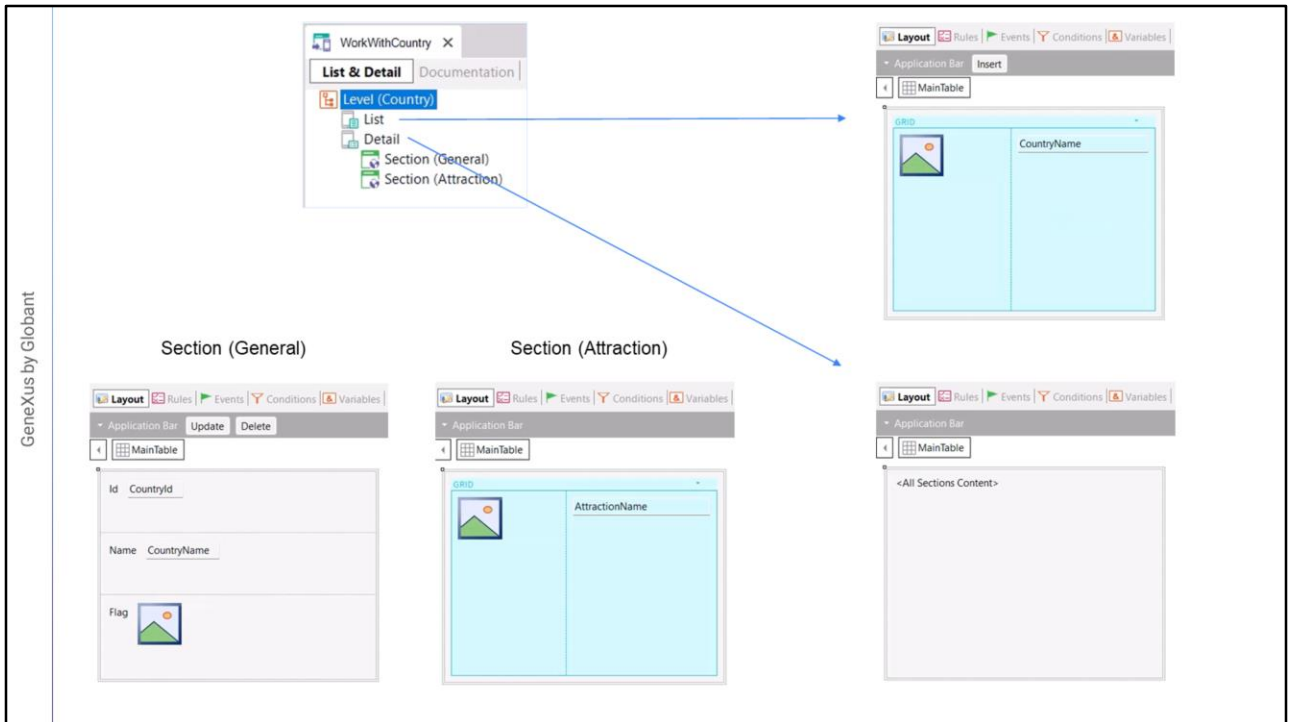
To test this we create a Menu object, set it as startup object, and as Item we set the WorkWithCountry object.

We run it.

We see the screens that were created from this pattern.

There is the list of countries, and if we select one we see its details. Here we have another tab that shows the attractions of that selected country; this is because each attraction will have an associated country, according to what we declare in its transaction. There will be as many tabs as transactions that have CountryId as foreign key –in this case only the Attraction transaction.

Returning to the detail of the selected country, we can edit or delete it.



Now let's see in GeneXus how everything is created.

When selecting the WorkWithCountry object, we see the two screens that will be generated:

The one for the list of countries.

And the one that corresponds to the detailed information of a country, with its two tabs, which are shown as "sections".

When selecting the List node, we see its Layout already initialized, with a grid that will show the value of the CountryName attribute of each country to be loaded from the database.

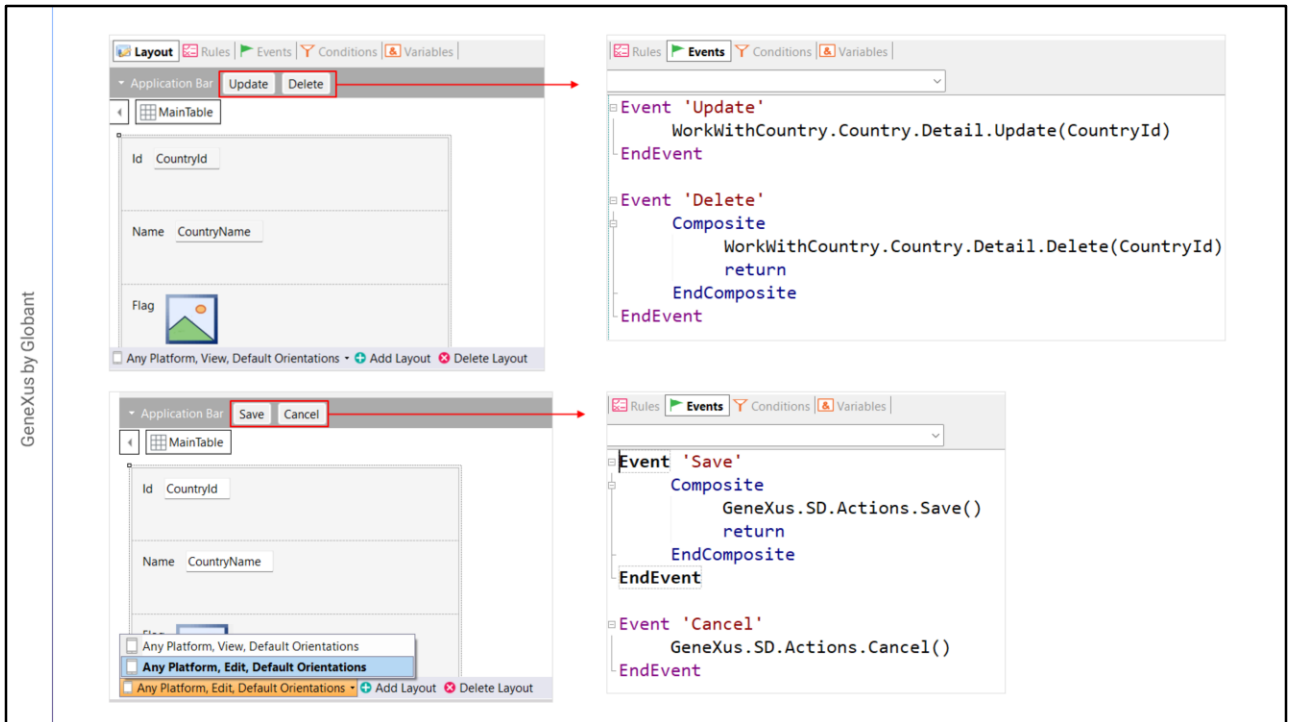
In addition, we see that sections are enabled to define rules, events, conditions and variables of this object.

If we open the Detail:

The screen object only contains a control called "<All Sections Content>". All the sections contained in this Detail will be loaded in this control. Here there are two: the one that shows the general information of the country with its attributes CountryId and CountryName.

And the one that shows the country's attractions in a grid, where the attributes AttractionPhoto and AttractionName are located.

The one that shows the detail of the country also has an application bar with two buttons: One to update and one to delete, as we saw in the running app.

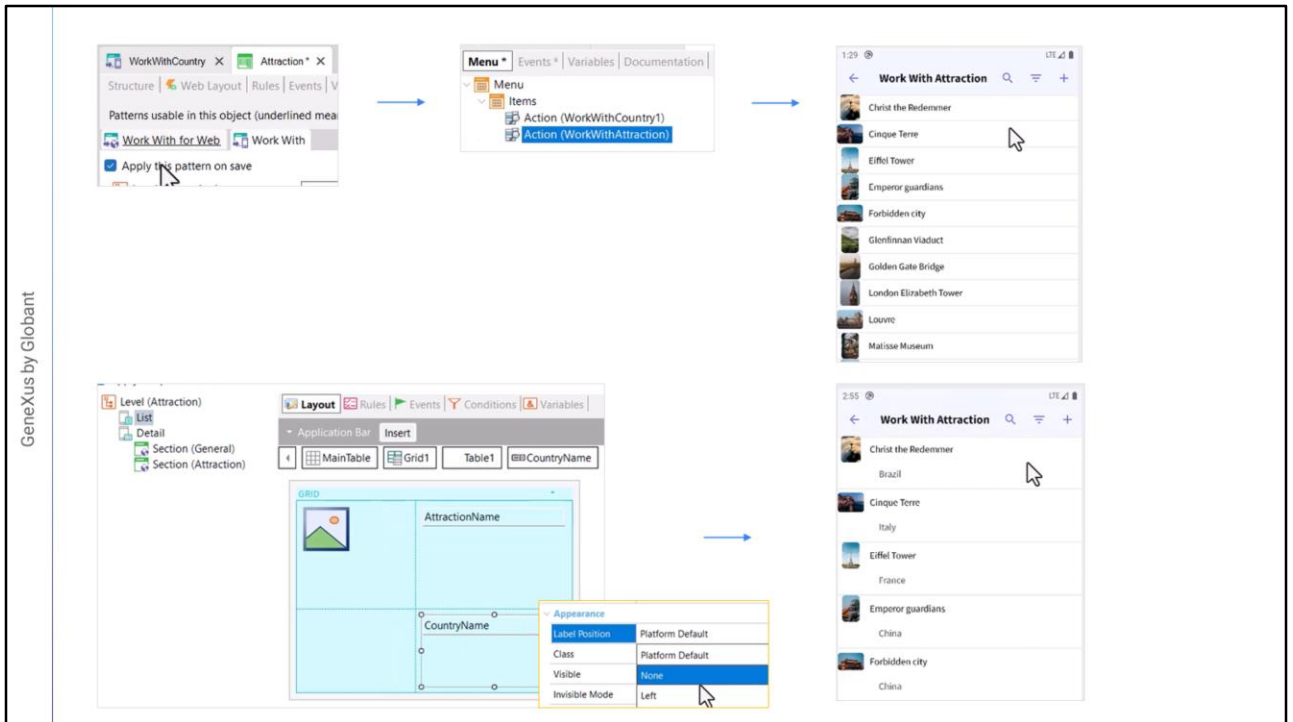


If we go to the events of these buttons, we see that one will take us to the screen in update mode, passing by parameter the ID of the country in which we are positioned, and the other will do it in delete mode, also passing the country's ID by parameter.

When we click on edit, the same Detail opens, but with another screen –the edit screen. It is initialized identically to the previous one, but with the option to edit the allowed fields, changing the application bar buttons to Save and Cancel, which in turn will also have their own particular events.

When the user clicks on the Save button, the device app will invoke –sending the data from this screen– the Rest service in the web server containing the transaction logic, and therefore will update the information on the database. All this is done in a transparent manner for the developer, who will not have to worry about these low-level details.

Returning to the list we see that the WorkWith also generated the option to create new records, in this case a country.

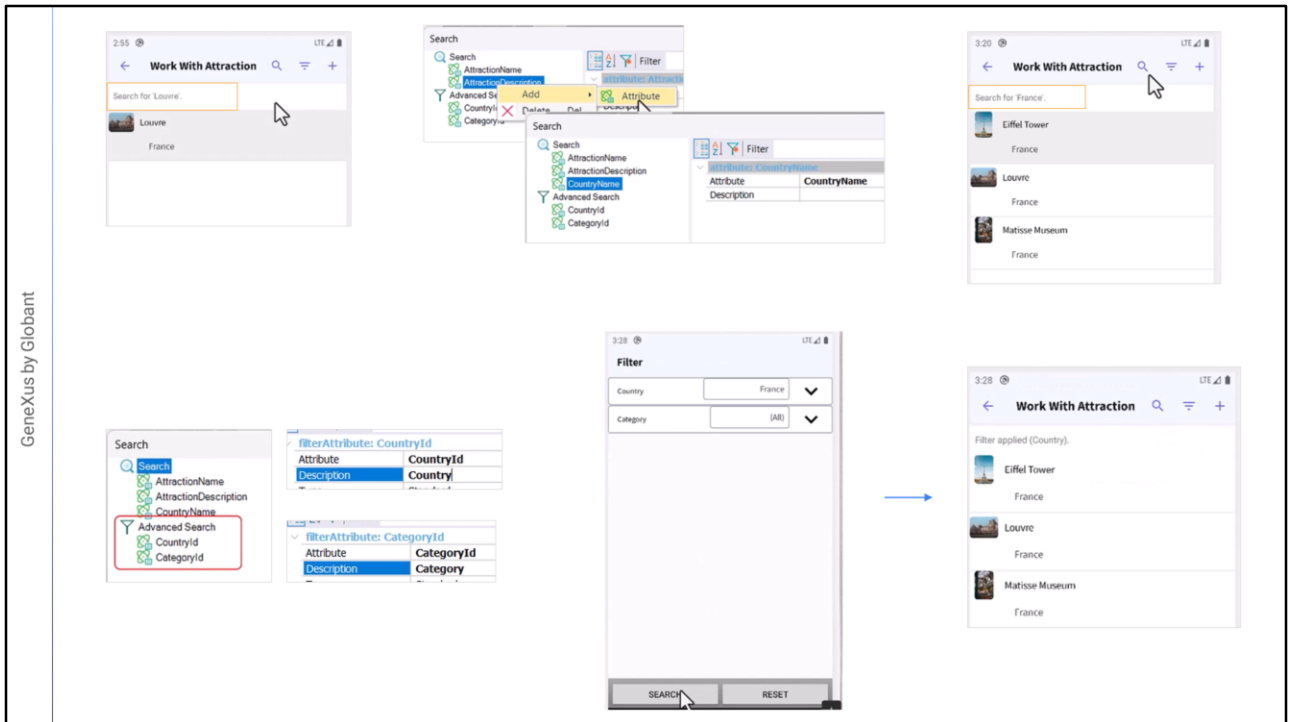


Now we will apply the WorkWith pattern also for the Attraction transaction, and once the corresponding object is generated, we add it as an item of the Menu object, to the WorkWith.

So, when we run the application again, we will be able to access the list of countries, or the list of attractions. We access the list of attractions.

The attraction's image and name are displayed in the grid. We may also want to show the country of the attraction.

To do so, we go to the attraction WorkWith, select the List node, go to the toolbox and drag the Attribute/Variable control, and choose the CountryName attribute. We will remove its Country label, change the Label Position property to None, and run again. Besides the name of the attraction, it now shows its country.



Note that by default it allows searching by attraction name. This is configured in the grid properties, in one named Search.

By default, the search is performed by attraction name and by description.

We could add CountryName, for example, to allow making searches by country name as well.

And then, searching for "France", the attractions of that country are displayed, since we filtered by the CountryName attribute that we just added.

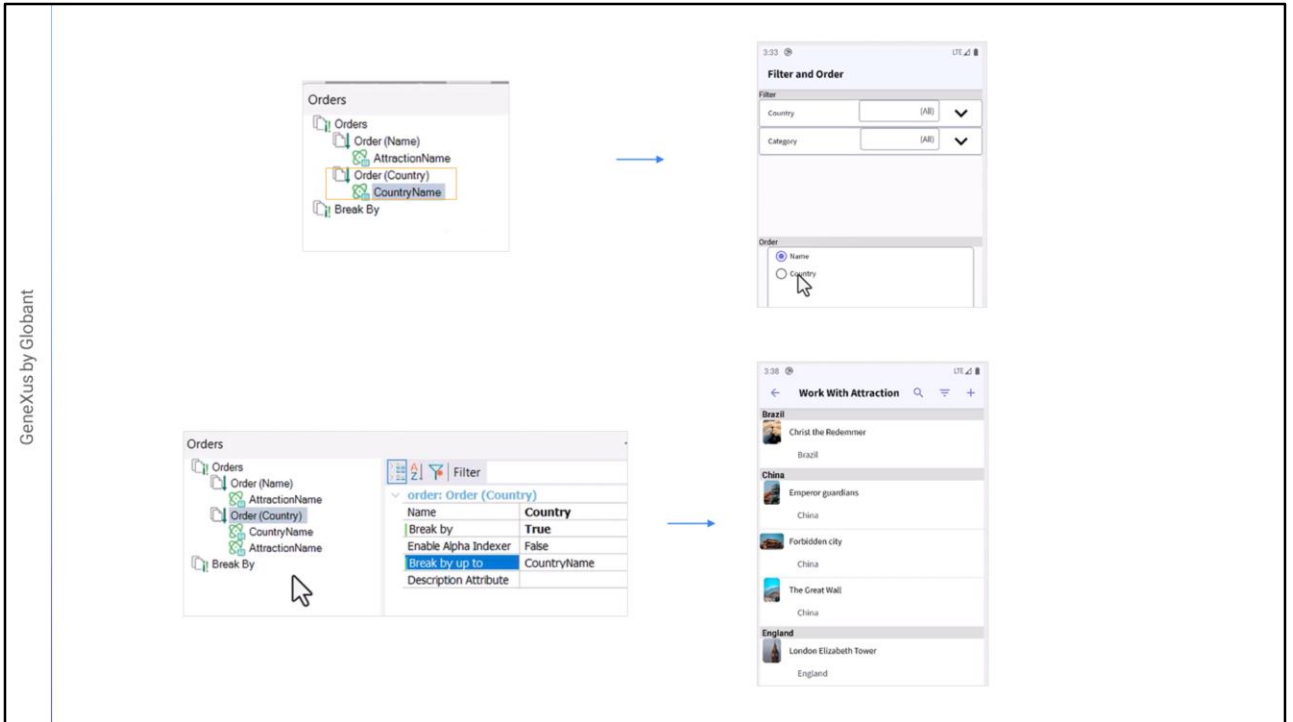
We also have advanced filters by the attributes CountryId and CategoryId, which, remember, are the foreign keys of the Attraction transaction.

This is also configured in the Search property, within the advanced search; from there we will be able to make the changes that we consider necessary.

We are going to change the descriptions by Country and Category, respectively, so that it is more intuitive, and execute again.

To filter the attractions by country, for example, it opens the country selection, so that we can select one of the existing ones. Selecting Search applies the filter, and the same with the category.

We can add or delete filters.



If we look at the list, it is ordered by AttractionName. This is specified in the order property of the grid.

We may, for example, want to give the user the option to sort by country.

For that we create a new order, and then we add the attribute by which we want to order, in this case CountryName.

When filtering, it also allows sorting by the different options that we have configured, now that we have two, it gives us the option to choose.

If we select by country, it now sorts alphabetically by country name.

Let's see one last example of what else we can modify here.

We may want this option to show the information grouped by country and for each country to sort by attraction name. To do this, we modify this order, adding the AttractionName attribute, and specifying that it will break the information by CountryName.

Let's see it running.

So far, we have seen a small sample of what the WorkWith pattern allows us to apply to transactions in mobile applications.

GX

GeneXus by Globant

GeneXus[™]
by **Globant**

training.genexus.com