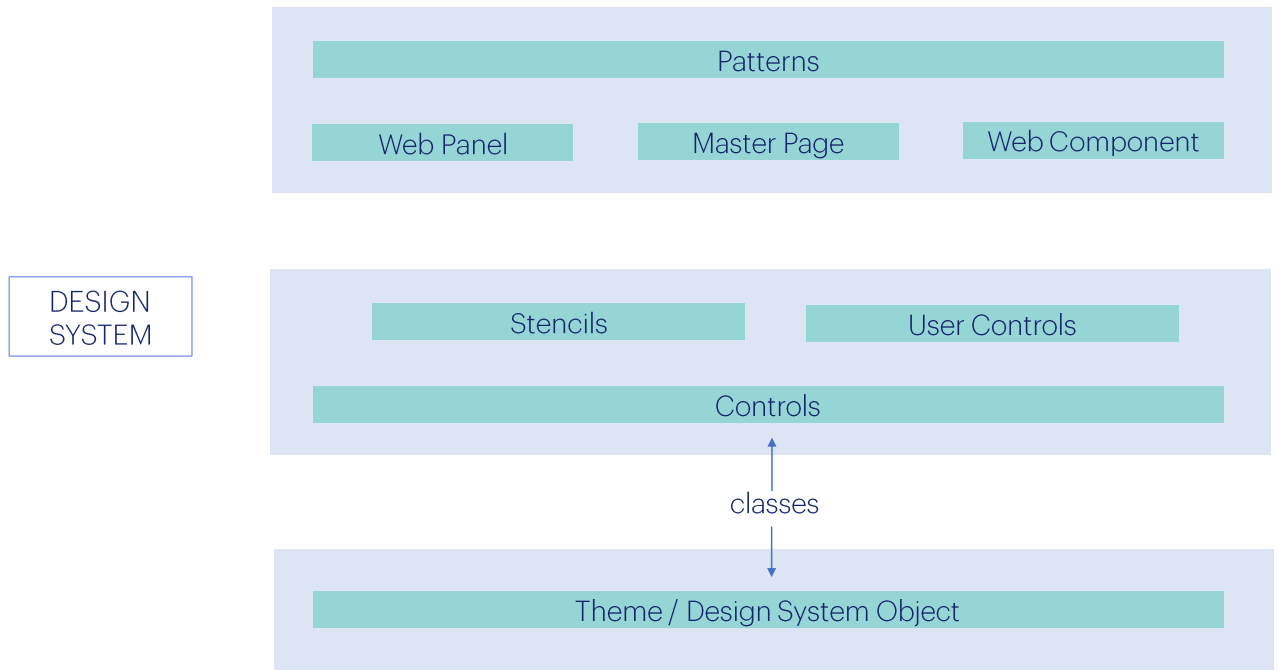


Design System in GeneXus

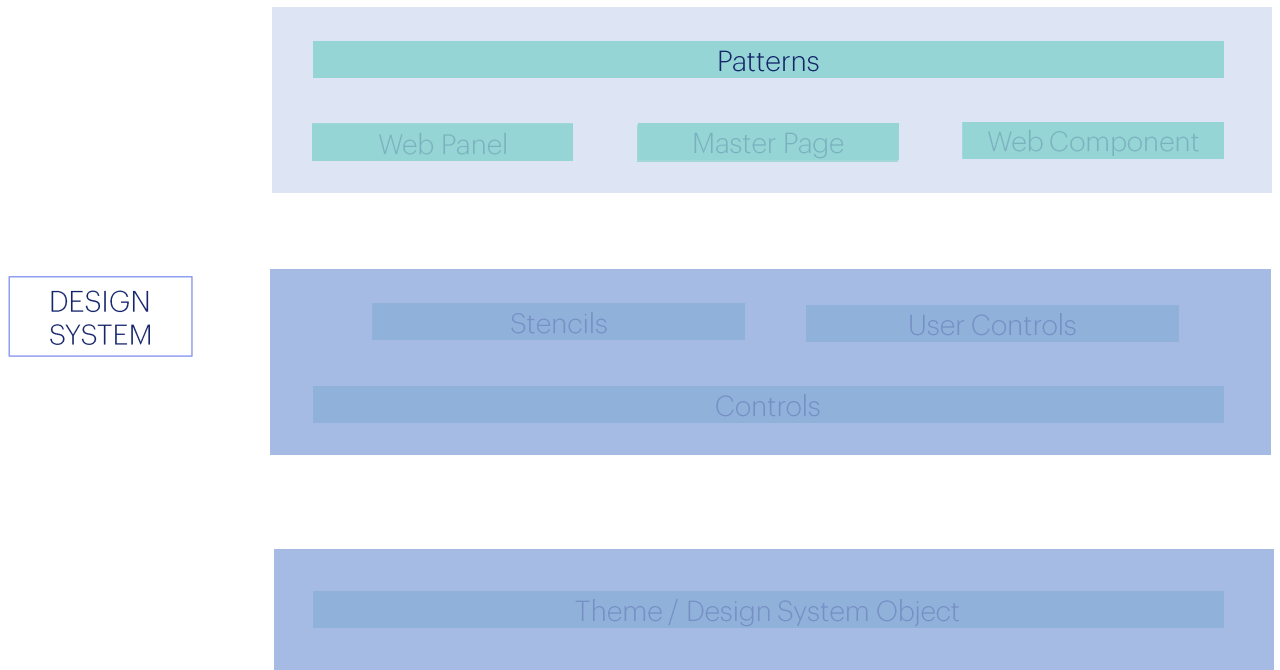
Overview





In GeneXus, which players take part in modeling the Design System of an application?

Here we will briefly describe them. After this course, you will be able to learn more about them if you want to focus on the User Interface of your application.



Using a top-down approach:








We have Patterns, such as the Work With of a transaction, which create Web Panel objects, among other things. These panels already include part of the Design System.

For example, actions have a common aesthetic, with a color that is also used in the buttons of the transaction or Web panel to display the information of an item. There is also a common header that all pages share by default. Not only those of the Work With, but also those of the Web Panels we created from scratch. To avoid repeating this common header in all Web Panels, there is a Master Page object.

Each web panel will be loaded inside the ContentPlaceHolder control of its Master Page. And here is that header we were looking at. When the Web Panel is executed, its Master Page is also executed and the page viewed in the Browser is the combination of both layouts, according to what the Master Page indicates.

Default Design System: Unanimo

The screenshot displays a web application titled "Travel Agency" using the GeneXus framework. The interface includes a dark sidebar menu with options: "Attractions", "Categories", and "Countries". The main content area, titled "Attractions", features a table with the following columns: "Id", "Name", "Country Name", "Category Name", "Photo", and "City Name". The table lists seven attractions, including "Christ the Redeemer", "Eiffel Tower", "Forbidden City", "Great Wall", "Louvre Museum", "Matisse Museum", and "Smithsonian Institute". To the right of the table, there are "UPDATE" and "DELETE" buttons for each row, which are highlighted by an orange box. Above the table, there is an "INSERT" button (also highlighted by an orange box) and a search bar labeled "Name". The browser's address bar shows the URL: "trialapps3.genexus.com/ld8424e2c2a0c10df40f0af431c97a22bd/wwattraction.aspx".

Id	Name	Country Name	Category Name	Photo	City Name	UPDATE	DELETE
4	Christ the Redeemer	Brazil	Monument		Rio de Janeiro	UPDATE	DELETE
3	Eiffel Tower	France	Monument		Paris	UPDATE	DELETE
7	Forbidden City	China	Monument		Beijing	UPDATE	DELETE
2	Great Wall	China	Tourist site		Beijing	UPDATE	DELETE
1	Louvre Museum	France	Museum		Paris	UPDATE	DELETE
6	Matisse Museum	France	Museum		Nice	UPDATE	DELETE
5	Smithsonian Institute	United States	Museum		Washington	UPDATE	DELETE

For example, we see that the actions have a common aesthetic, with a color that is repeated as well....

Default Design System: Unanimio

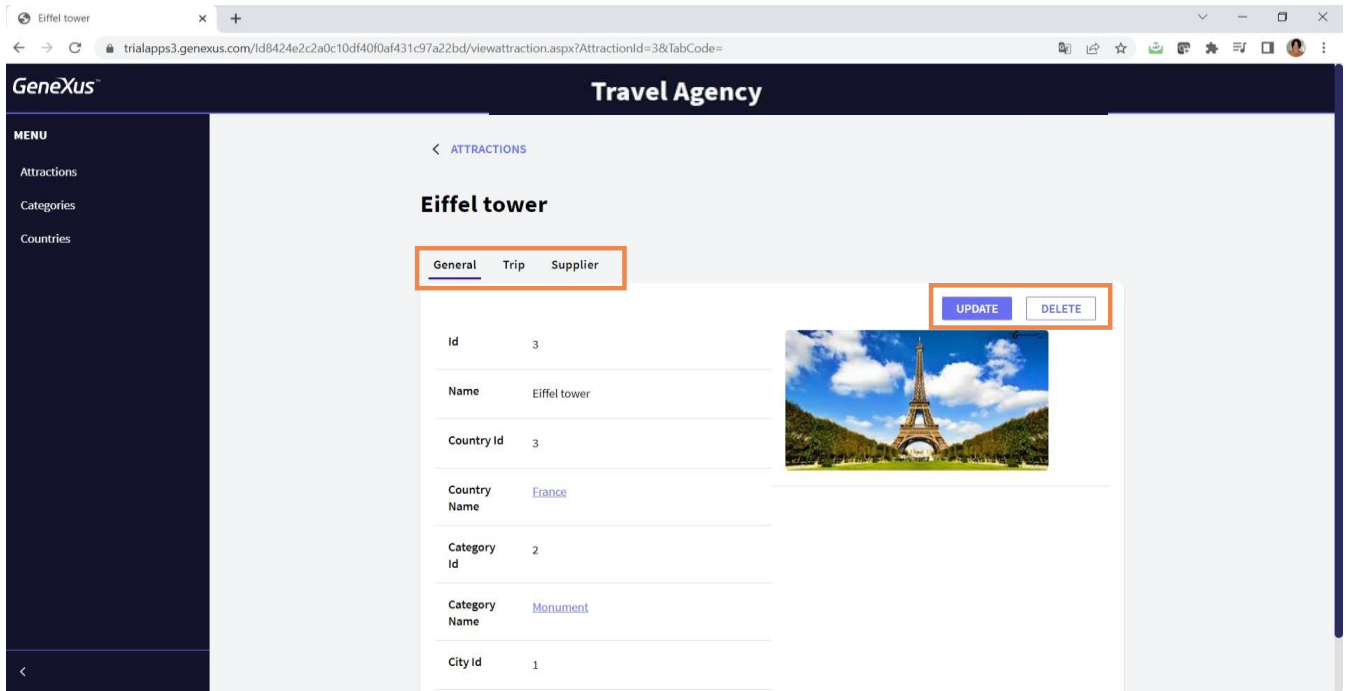
The screenshot displays a web browser window with the URL `trialapps3.genexus.com/ld8424e2c2a0c10df40f0af431c97a22bd/attraction.aspx?Mode=INS&AttractionId=0`. The application is titled "Travel Agency" and features a dark blue sidebar menu with the following items: "MENU", "Attractions", "Categories", and "Countries". The main content area is titled "Attraction" and contains a form with the following fields:

- Id**: 0
- Name**:
- Country Id**: ☒
- Country Name**:
- Category Id**: ☒
- Category Name**:
- Photo**: ☒
- City Id**: ☒

At the bottom right of the form, there are two buttons: "CONFIRM" and "CANCEL". The "CONFIRM" button is highlighted with an orange border.

... in the transaction buttons or...

Default Design System: Unanimo



... those of the Web panel to view the information of an element.

There is also a common header that all pages share by default. Not only those of Work With...

Default Design System: Unanimo - MasterPage

Web Panel5

trialapps3.genexus.com/Id8424e2c2a0c10df40f0af431c97a22bd/webpanel5.aspx

GeneXus **Travel Agency**

MENU

- Attractions
- Categories
- Countries

Country Id: (None)

Attraction Name From:

Attraction Name To:

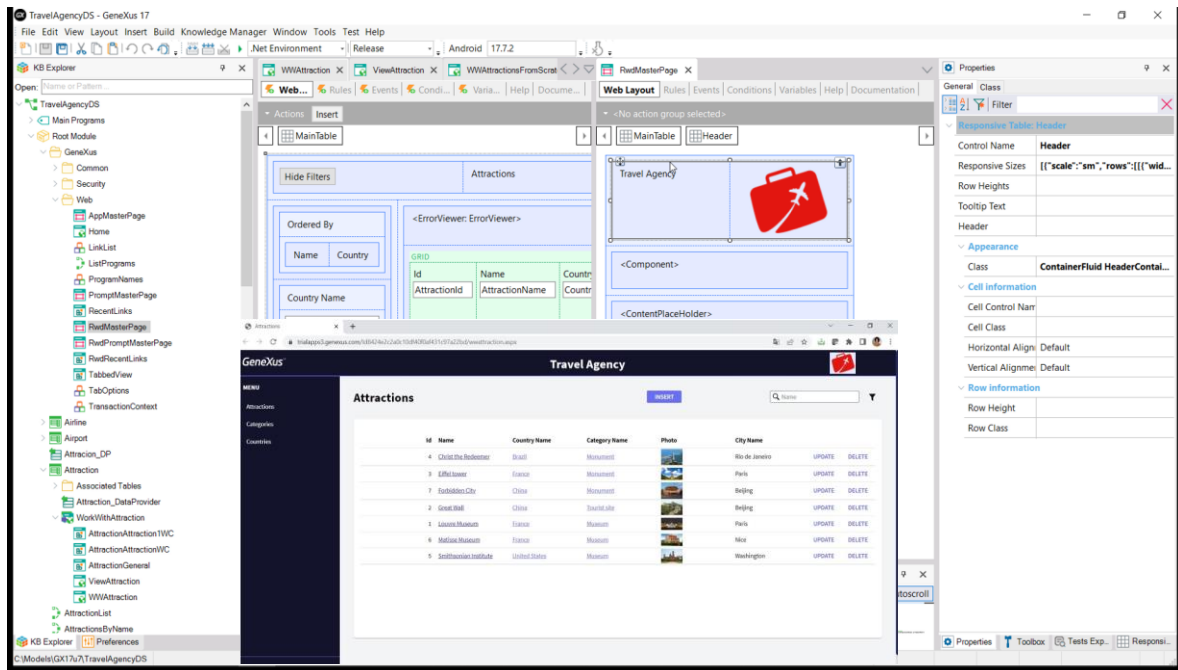
Attraction Name	Country Name	Attraction Photo	Trips
Christ the Redeemer	Brazil		2
Eiffel tower	France		2
Forbidden City	China		0
Great Wall	China		0
Louvre Museum	France		0
Matisse Museum	France		1
Smithsonian Institute	United States		1

Total Trips: 6

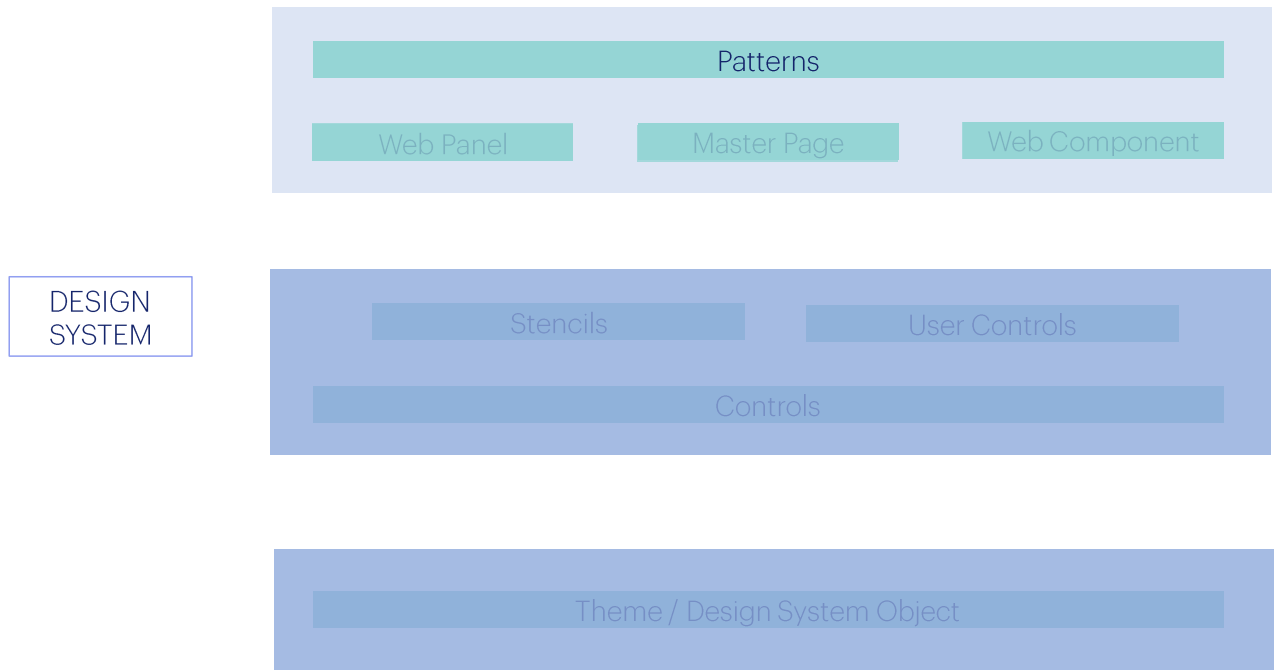
... but also those of the Web Panels we created from scratch.

To avoid repeating this common header in all Web Panels, there is a Master Page object.

Default MasterPage: MasterUnanimoslidebar



Each web panel will be loaded inside the ContentPlaceHolder control of its Master Page. And here is that header we were looking at. When the Web Panel is executed, its Master Page is also executed and the page viewed in the Browser is the combination of both layouts, according to what the Master Page indicates.



But there is also another way to compose panels. They are Web Component objects; that is, they are like small pieces of Web panels that can be inserted into other Web Panels.

For example, the Work With pattern already did it.

Pattern Work With: Web Component

GeneXus™ Travel Agency


MENU

- Attractions
- Categories
- Countries

< ATTRACTIONS

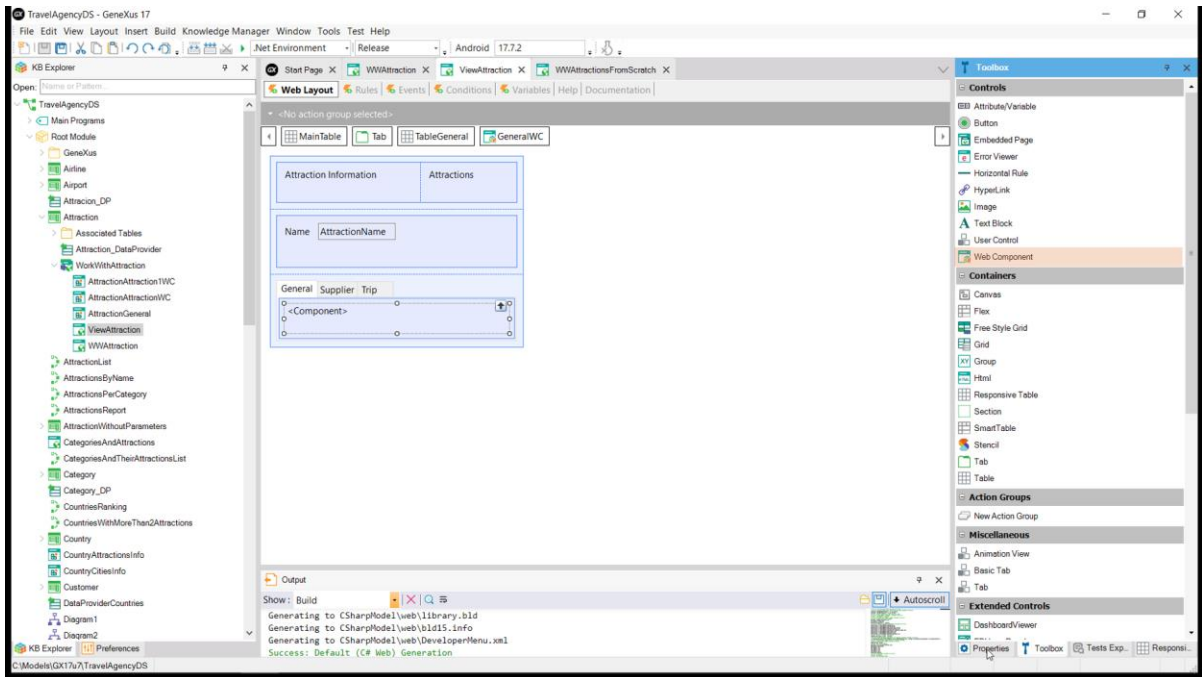
Louvre Museum

General Trip Supplier

Id	1	 UPDATE DELETE
Name	Louvre Museum	
Country Id	3	
Country Name	France	
Category Id	1	
Category Name	Museum	
City Id	1	

In the Web Panel showing a tourist attraction, we see three tabs, the first one showing the general information of the attraction.

Pattern Work With: Web Component



In GeneXus, for the General tab we don't have attribute controls, but a control of component type.

Pattern Work With: Web Component

The screenshot displays the GeneXus IDE interface. The main editor shows the 'Events' tab for the 'Load Tab' event. The code logic is as follows:

```

12: When none
13:   Form.Caption = "Record not found"
14:   ViewAll.Visible = false
15:   &Exists = false
16: Endfor
17: &LoadAllTabs = false
18:
19: If &Exists
20:   &SelectedTabCode = &TabCode
21:   Tab.ActivePageControlName = &SelectedTabCode
22:   Do 'Load Tab'
23: Endif
24: EndEvent
25:
26: Event Tab.TabChanged
27:   &SelectedTabCode = Tab.ActivePageControlName
28:   &LoadAllTabs = false
29:   Do 'Load Tab'
30: EndEvent
31:
32: Sub 'Load Tab'
33:   if &LoadAllTabs or &SelectedTabCode = '' or &SelectedTabCode = "General"
34:     GeneralIWC.Object = AttractionGeneral.Create(&AttractionId)
35:   endif
36:   if &LoadAllTabs or &SelectedTabCode = "Attraction"
37:     AttractionIWC.Object = AttractionAttractionIWC.Create(&AttractionId)
38:   endif
39:   if &LoadAllTabs or &SelectedTabCode = "Attraction1"
40:     AttractionIWC.Object = AttractionAttractionIWC.Create(&AttractionId)
41:   endif
42: EndSub
43:

```

The Properties panel on the right shows the configuration for the 'Web Component: AttractionGeneral'.

Property	Value
Name	AttractionGeneral
Description	Attraction General
Style	Carmine
Type	Component
URL access	No
Show Master Page	False
Main program	False
On session timeout	Ignore
Focus control	Use Environment property va...
Cache expiration la...	
Automatic refresh	Yes
Auto compress http	Use Environment property va...
Qualified Name	AttractionGeneral
Object Visibility	Public
Warning messages	Disabled warning: spc0096 spc0107 spc0142
Compatibility	Standard Function: Only standard functions
Web interface	Web User Experie: Smooth
Datepicker	Enable Datepick: Use Environment property va...
Show week num...	Use Environment property va...
First day of wee...	Use Environment property va...
Web information	

The Output panel at the bottom shows the build process:

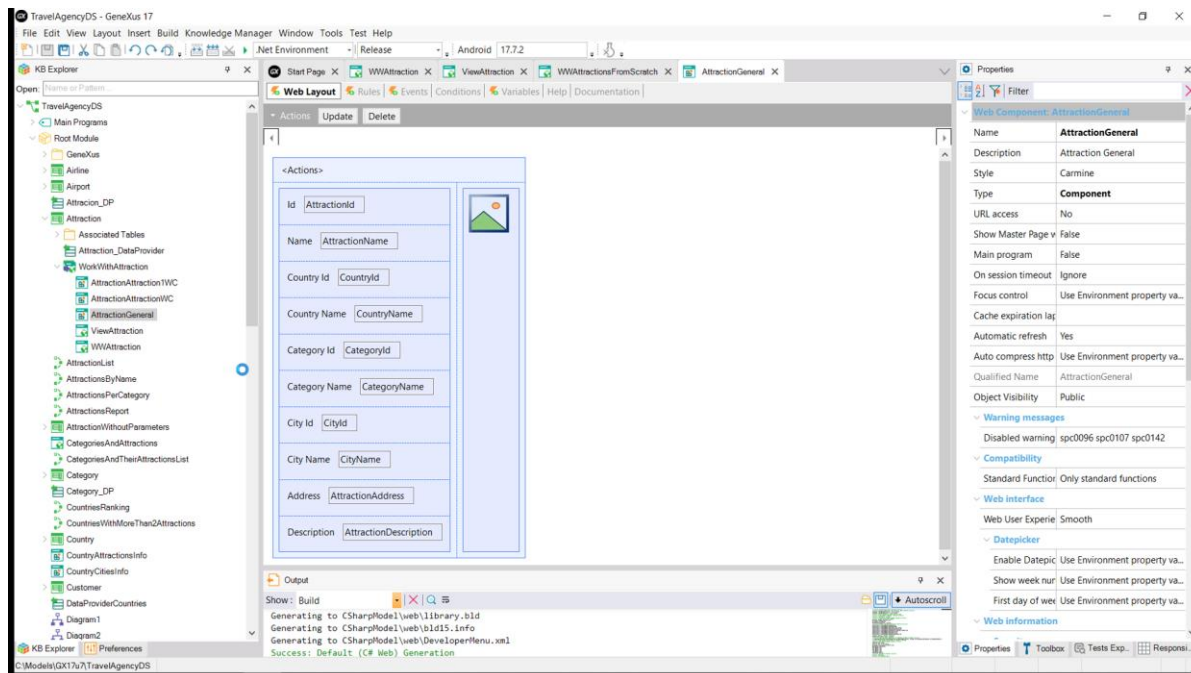
```

Show: Build
Generating to C:\SharpModel\web\library.bld
Generating to C:\SharpModel\web\bld\5.info
Generating to C:\SharpModel\web\DeveloperMenu.xml
Success: Default (C# Web) Generation

```

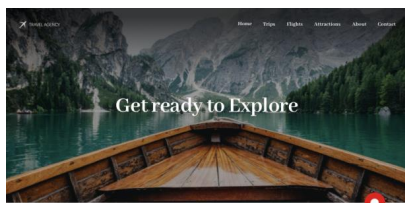
And in the events it is indicated that a Web Component should be loaded in there. The one named AttractionGeneral. We see that it is of Component type.

Pattern Work With: WebComponent



If we open it, it looks like an ordinary web panel, with its parm rule and events.

The pattern has already built its screens with these three objects, in order to reuse as much as possible. We should also use them to create high-quality, cost-effective systems that can be implemented once and reused as many times as necessary.



The new age of EXPLORATION

CONTACT US TODAY

At Travel Agency, we have consultants with an average of 10 years experience and a passion for travel activities that will work out the details and create unforgettable vacations.

We will work with you to plan a worry-free adventure that meets your travel needs, expectations and budget. When you plan your vacation with us we are there throughout the entire process. The more exciting possibilities available to you before, during and after travel.

We plan river and coast cruises, romantic honeymoon packages to sunny destinations, family, adventure and wellness travel along with international and domestic customized itineraries.

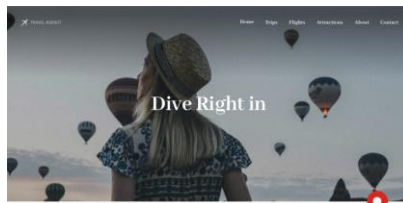


Let us help you plan the perfect trip

0123456789

© All rights reserved

© All rights reserved

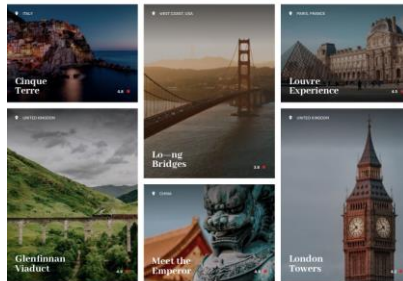


Popular ATTRACTIONS

CONTACT US TODAY

Travel Agency selects the most popular attractions around the globe in order to give you the most amazing the experience.

We are constantly searching for new adventures to offer with customer's needs. We work with a wide variety of suppliers from more substantial destinations to budget-friendly cities around the world.



Let us help you plan the perfect trip

0123456789

© All rights reserved

© All rights reserved



Visiting the LOUVRE MUSEUM

CONTACT US TODAY

Visit the palace of French Kings to witness some of the world's finest art. The Louvre holds many of Western Civilization's most famous masterpieces, including the Mona Lisa by Leonardo da Vinci, and a line of the top things to do in Paris.

A large number of the museum's paintings were created by the master Michelangelo. In the center of the Louvre Museum, other artists were so inspired through French master with the Louvre and the Museum's history, and the collection was further enriched by the spirit of Napoleon.

The museum holds 60,000 artworks over a 60,000 square meter exhibition space in three sections: the Denon, Sully, and Richelieu wings. Each wing has more than 70 rooms displaying paintings and objects of art, which there are numerous ways that each collection. The Louvre is one of the most visited museums in the world. To get the most out of your time spent at the Louvre, there are a number of excellent art books using this guide.

Exhibitions



Let us help you plan the perfect trip

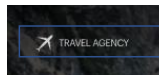
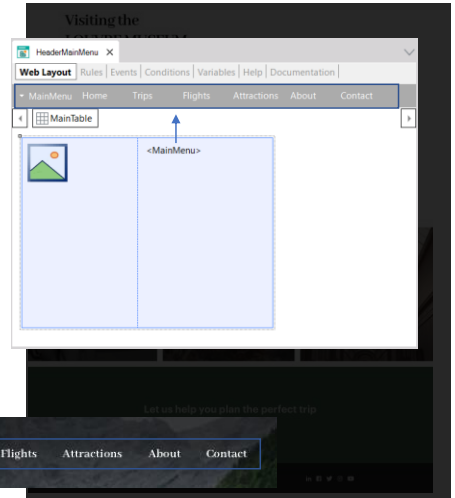
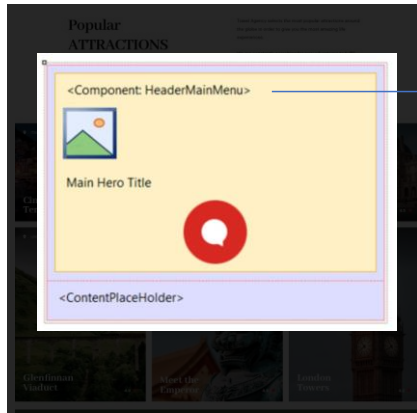
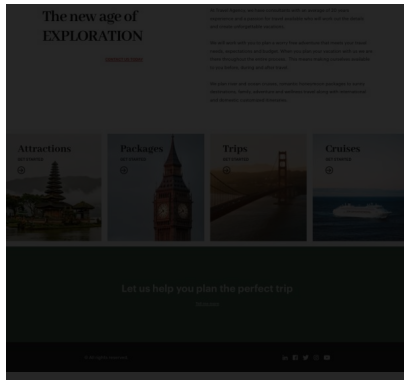
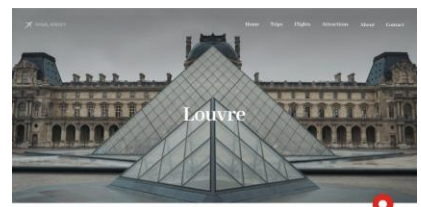
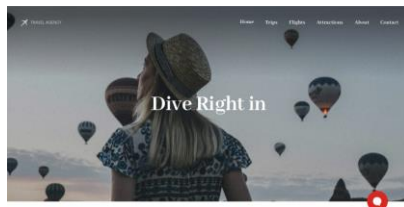
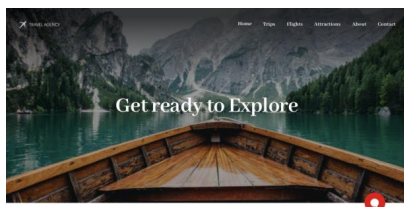
0123456789

© All rights reserved

© All rights reserved

For example, suppose that now we are not interested in the back-office application that uses the pattern, but in the customer-facing one, i.e. the one that will be used by the end customers.

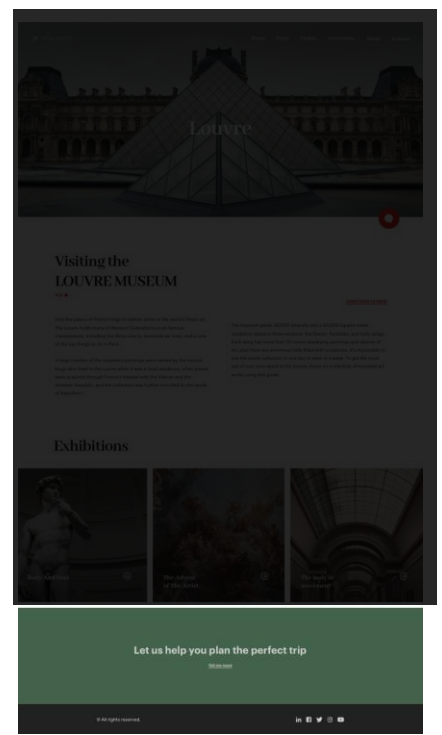
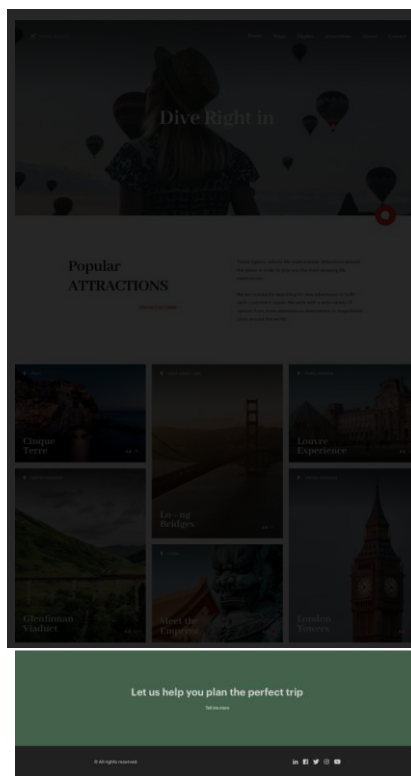
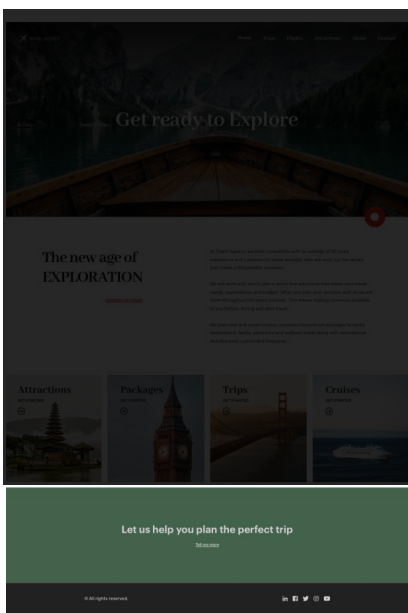
Suppose that the designers of our team designed the first three screens of the application. The Home, from which you call one that shows all the tourist attractions that can be visited; also, from this one, by choosing an attraction, the last one is called, which shows the information of that chosen tourist attraction.



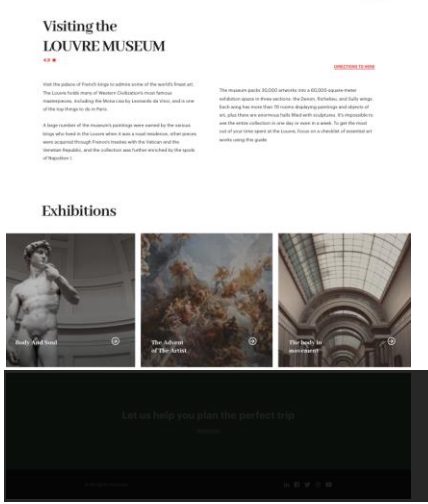
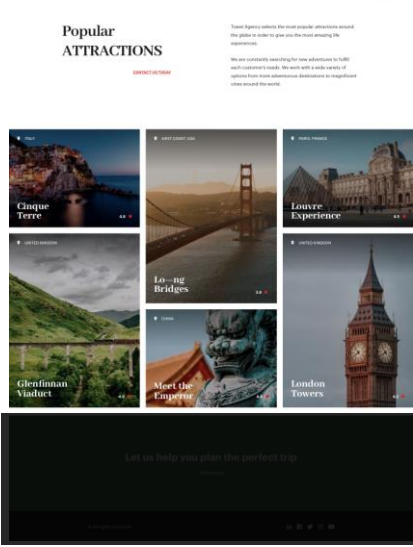
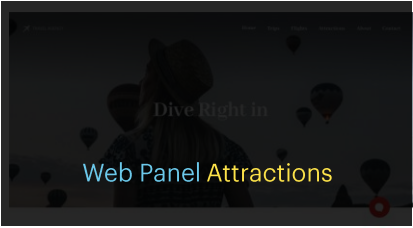
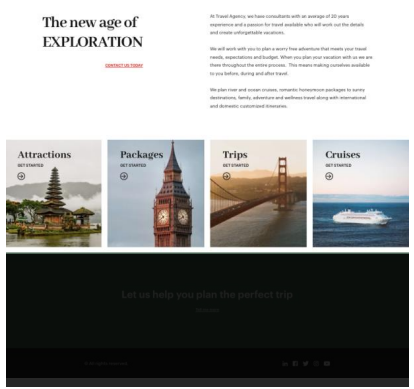
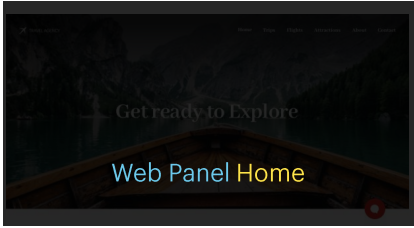
Note that from a design point of view there are repetitions: the three screens have a header with a background image, an overlay text, a menu and a logo, and an image representing a chatbot. Ideally, a Master Page should be used to implement this common header, where, depending on the object being loaded in the content container, the background image and text will be displayed (this is programmed in the Start event of the Master Page).

If we need to reuse the logo and menu somewhere else, we can choose not to implement them directly in the Master Page itself, but in a Web Component, which we insert in the Master Page... and everywhere else we need it, of course.

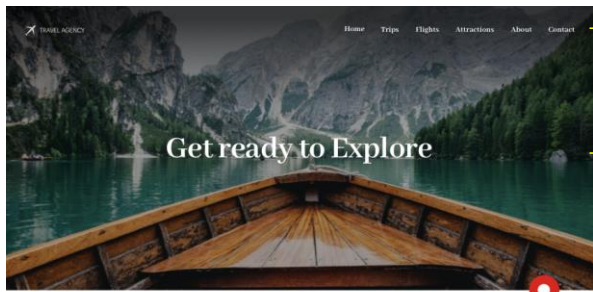
The web component has behavior: for example, we must program which object to call for each menu action. We also want to program an action when the user clicks on the logo. That's why it is not an object that allows reusing only one part of the screen, but rather it can be considered as a miniature Web Panel. For most purposes it is a Web Panel.



OK, let's go on... We didn't say it, but, of course, the common footer should also be placed in the Mater Page.

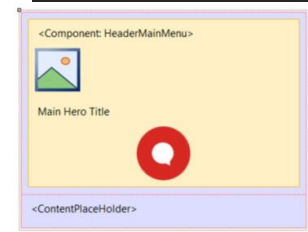
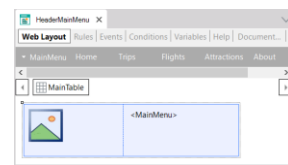


The specific content of each page will be implemented in each individual Web Panel.



Web Component

Master Page



The new age of
EXPLORATION

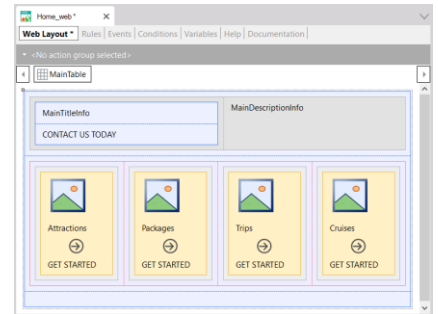
[CONTACT US TODAY](#)

At Travel Agency, we have consultants with an average of 20 years experience and a passion for travel available who will work out the details and create unforgettable vacations.

We will work with you to plan a worry free adventure that meets your travel needs, expectations and budget. When you plan your vacation with us we are there throughout the entire process. This means making ourselves available to you before, during and after travel.

We plan river and ocean cruises, romantic honeymoon packages to sunny destinations, family, adventure and wellness travel along with international and domestic customized itineraries.

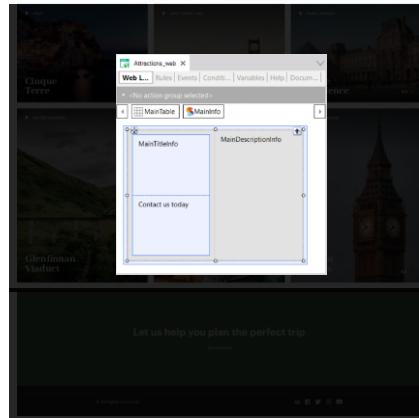
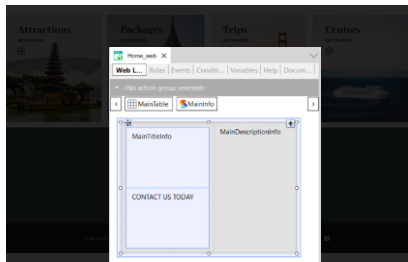
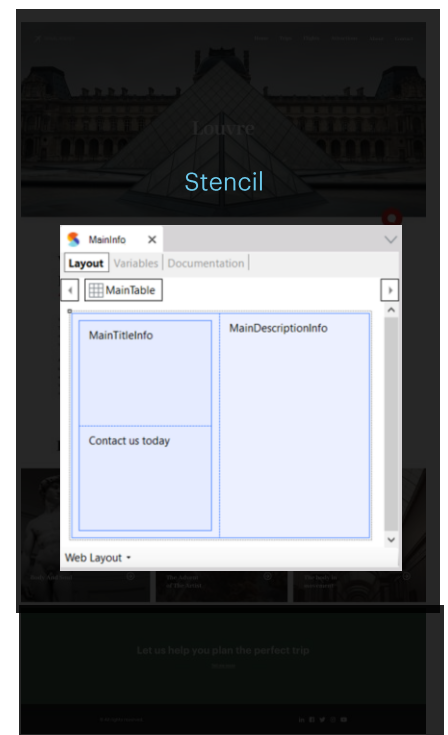
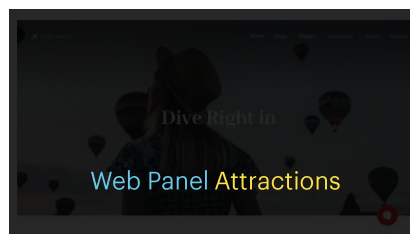
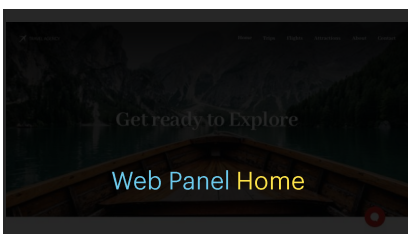
Web Panel



These objects (Web Panel, Master Page, and Web Component) are executable; that is, they have behavior.

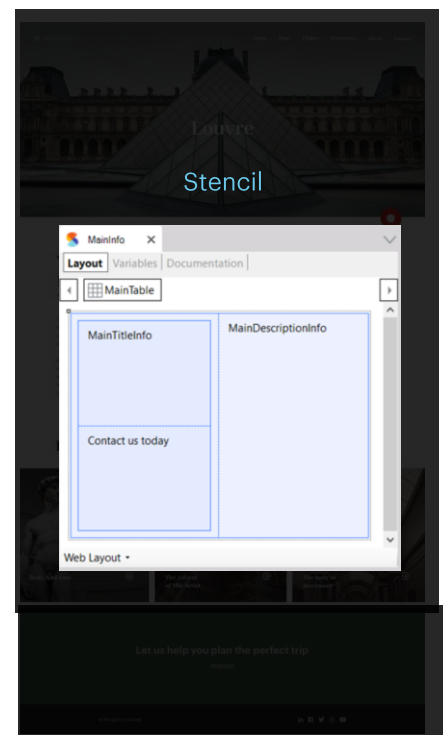
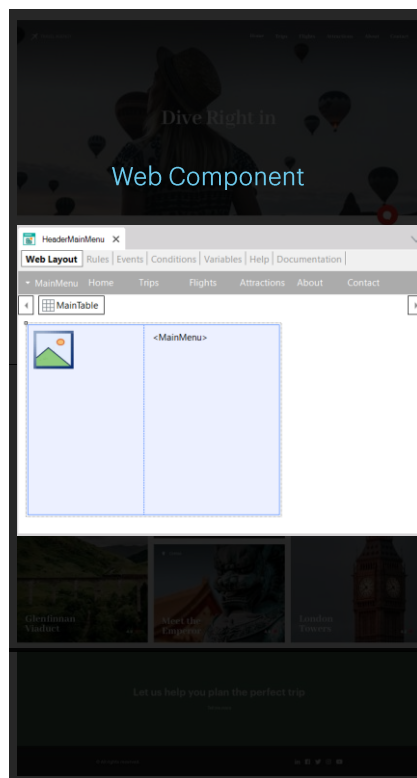
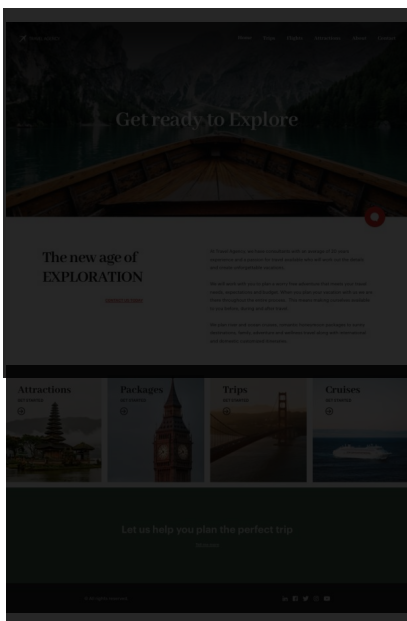
But there is also another type of object that does offer to reuse only a piece of layout, without behavior. It is the Stencil object, which allows doing the same thing that Web Components do, but only at screen level, so it will not have rules or events.

It is a way to build more complex controls from simpler controls.



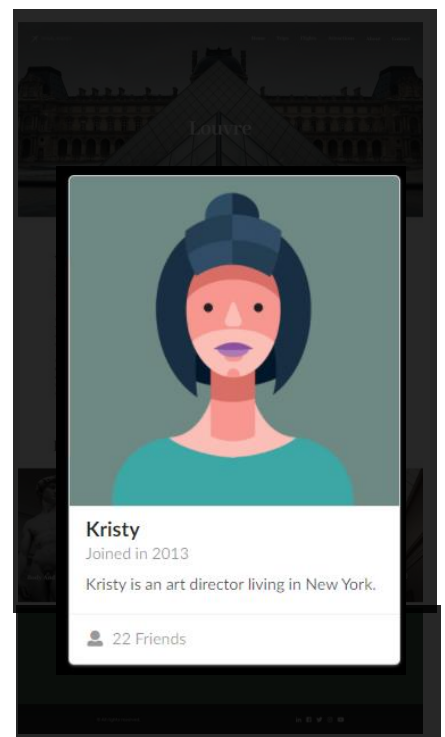
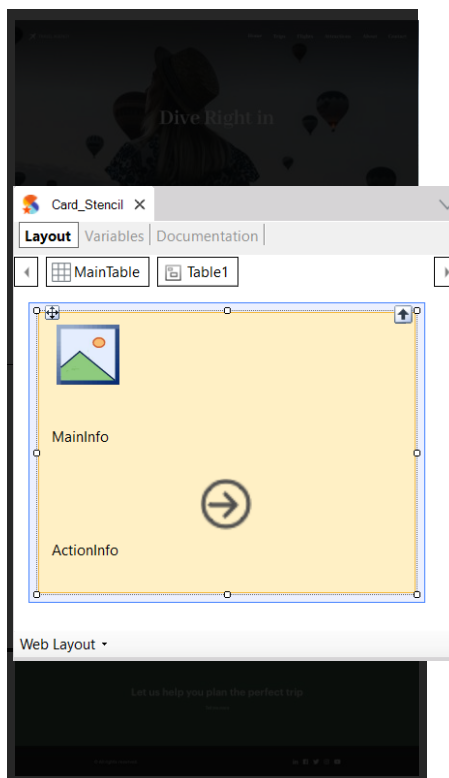
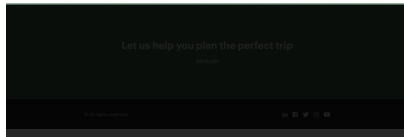
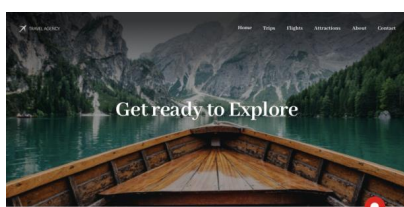
If we look at the first two Web Panels, we see that this screen section has the same structure and design. In both Web panels, we would have to repeat tables with text blocks and give them exactly the same layout. To avoid these repetitions, we can group these controls in a Stencil object, and then insert a stencil control in both Web Panels, which is loaded with the created stencil object.

In general, if we know that the structure and design of a set of controls on a screen will be repeated on many screens, even if they have different content, then we can group them in a Stencil object and use that object in the Web Panel, Master Page, or Web Component.



The main difference between a Web Component and a Stencil is that the Stencil has no behavior; that is, it is not an executable object, while the Web Component is.

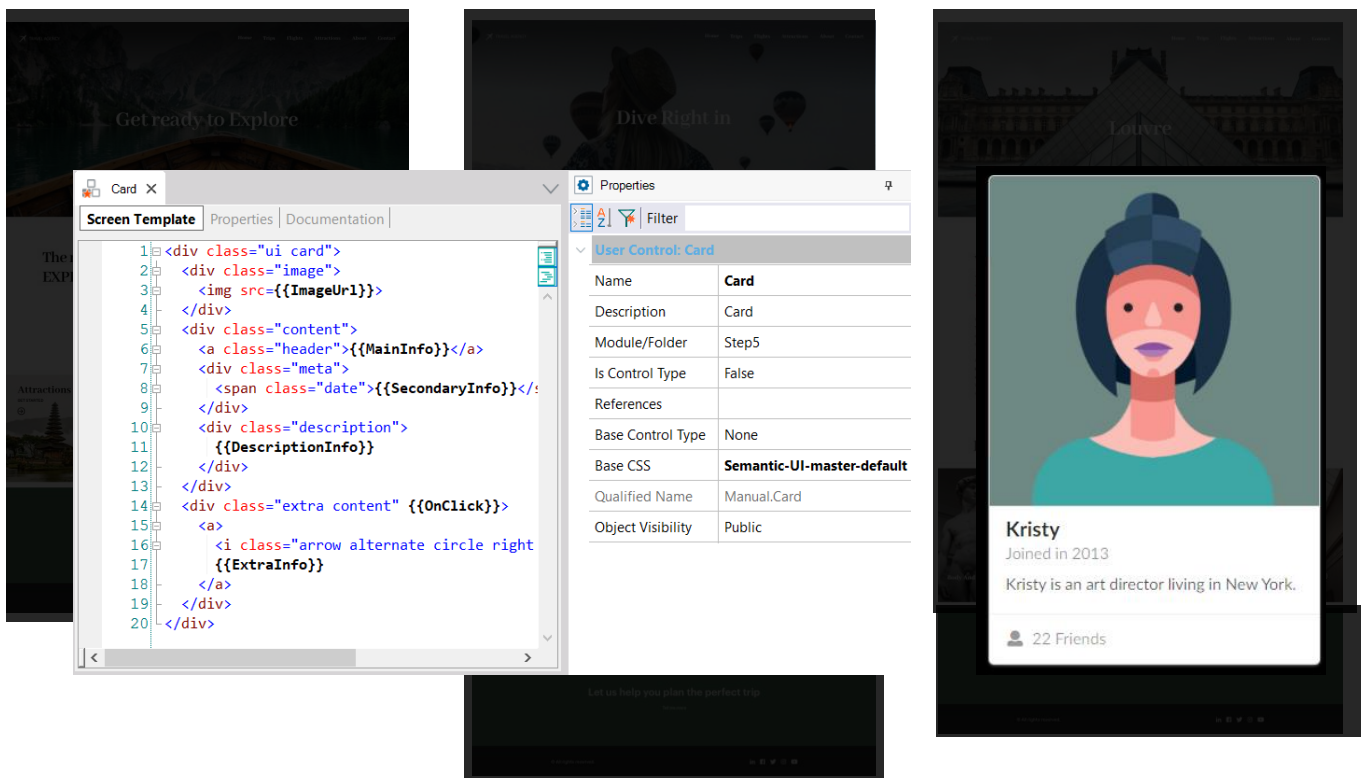
This is why the Stencil is clearly a Design object. It is entirely intended for the Design System.



Another example of use of Stencils: we want to implement the menu with images that we see on the home page. There are 4 options with identical design. They are like cards. Then we could create a Stencil and reuse it 4 times.

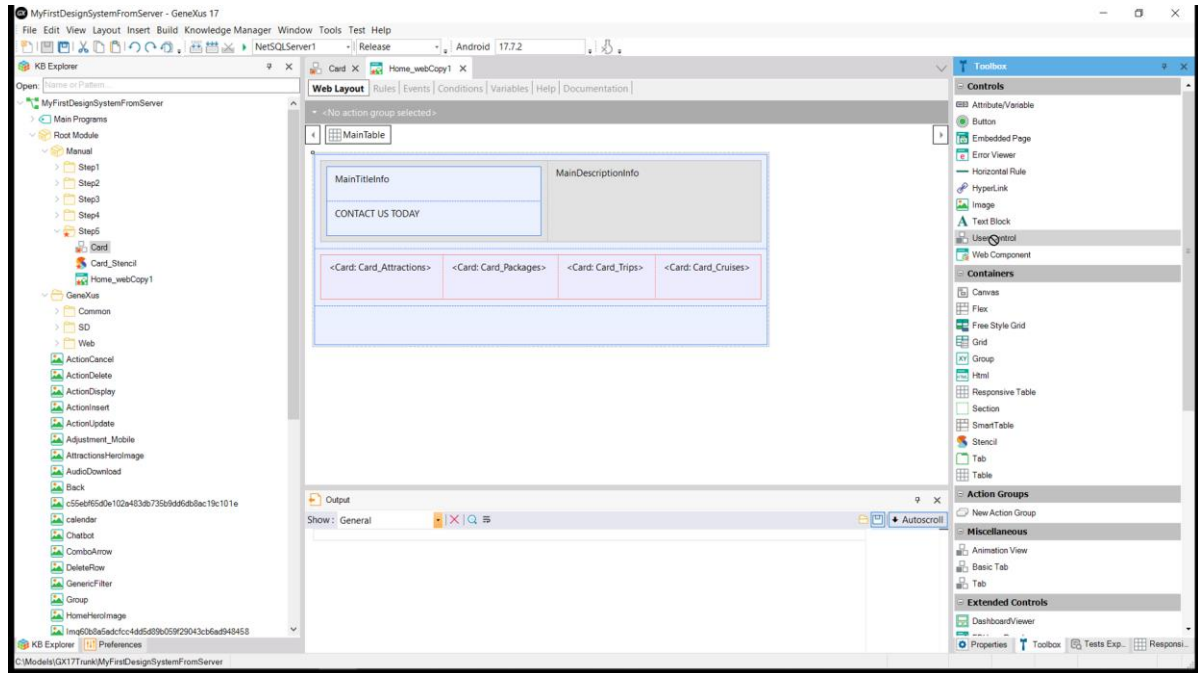
In this Stencil, we would have to overlay two texts and an arrow image on a background image in order to create this kind of card. That is, we are composing a kind of larger control using tables, images, variables, textblocks. And the question that we will answer in a moment is how we design all these controls.

The other option, without using a stencil, is to directly incorporate a control made by a third party. For example, the control to design cards provided by the SemanticUI design framework.



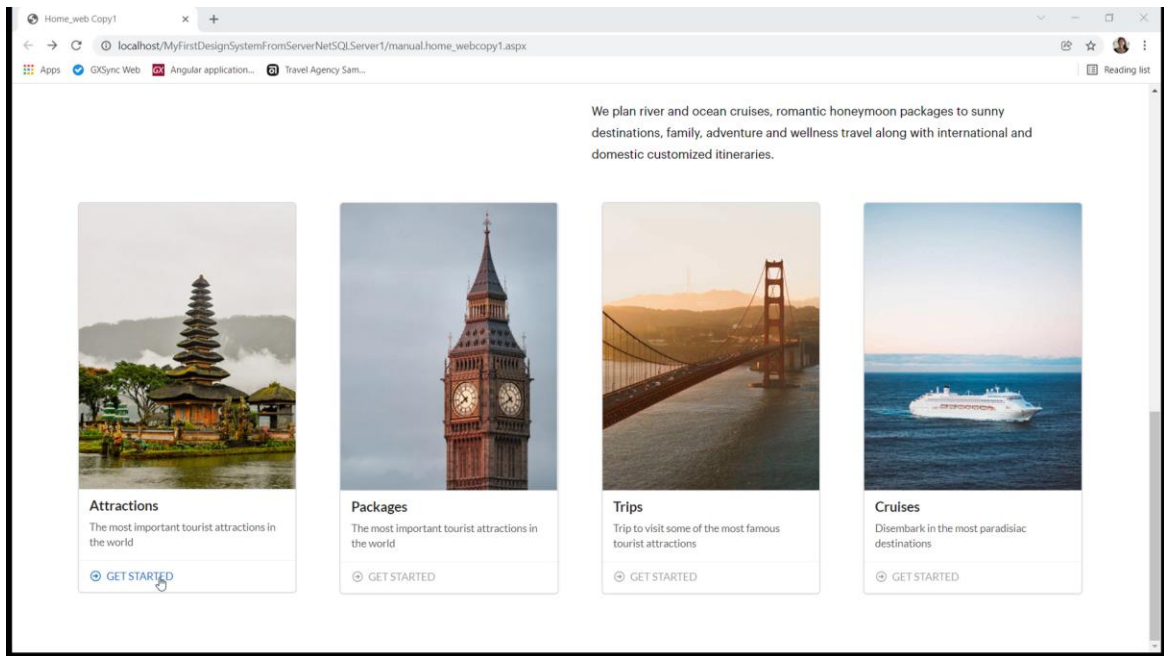
To do so, it is enough to create a User Control object in GeneXus. Then, copy the HTML code provided by the supplier, change the fixed data to variable data, and indicate the file containing the style of the HTML elements. That is, the CSS file provided by the supplier.

User Control

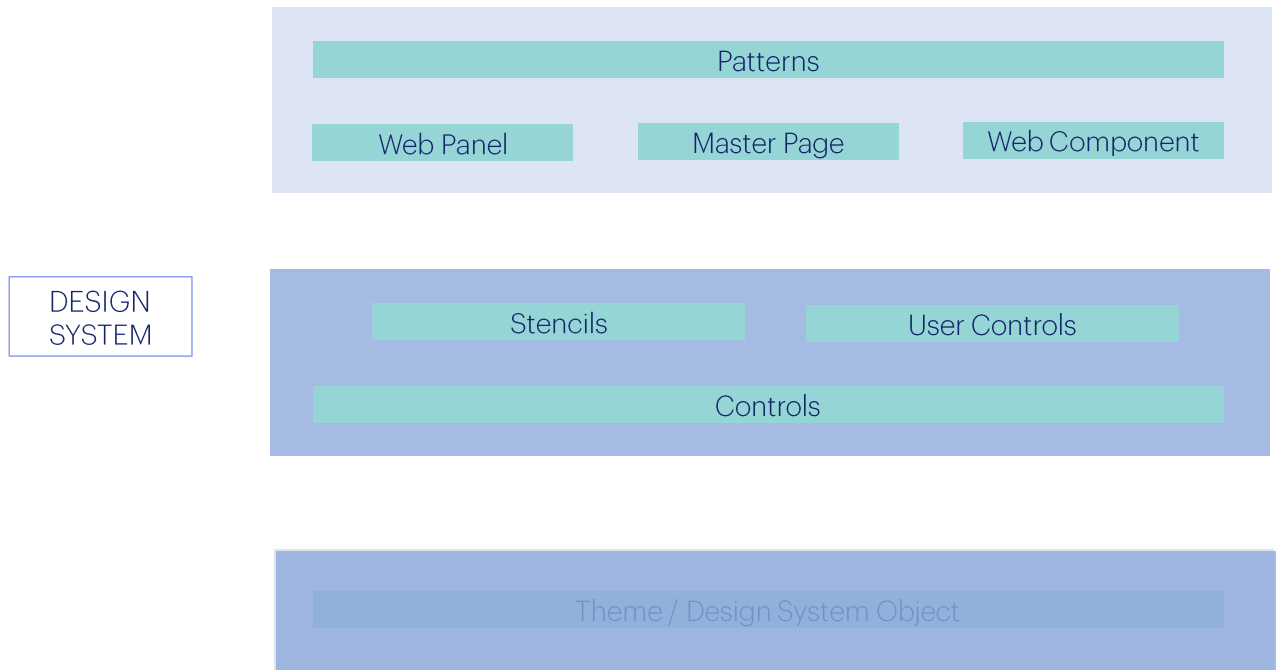


After doing this, we can use it as another control from the toolbox.

User Control



And so, at runtime we will see...



In short: first we talked about the patterns that use Web Panels, Master Page, Web Components to implement parts of the application.

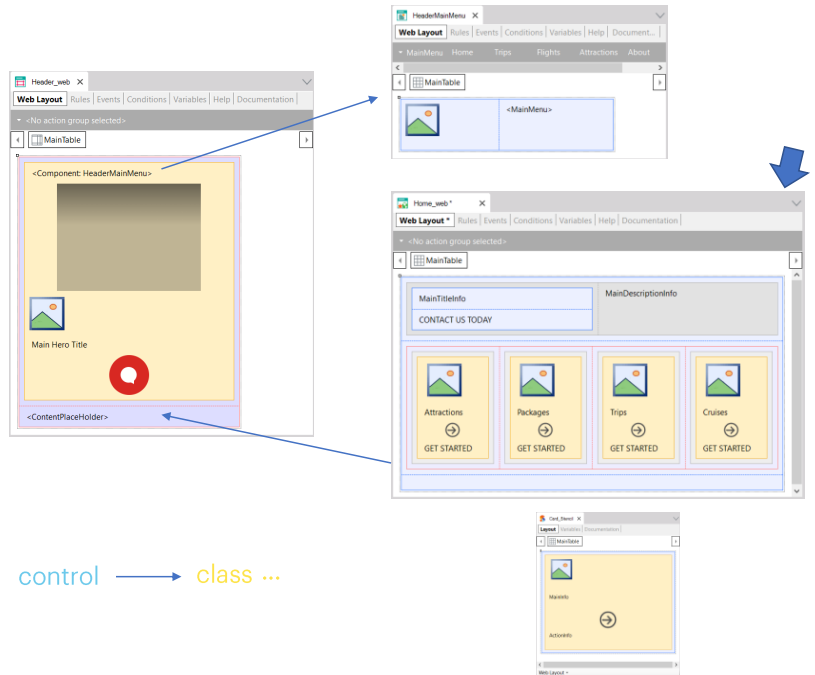
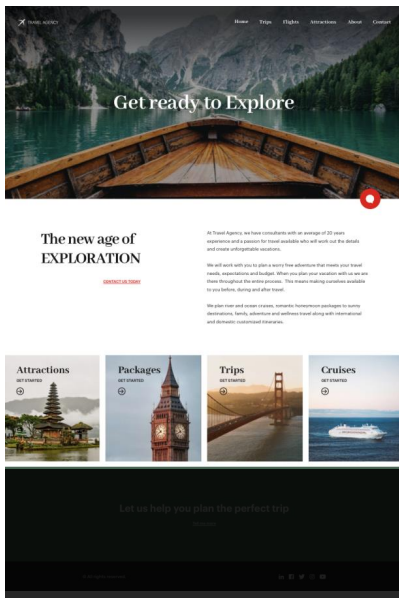
We've seen these objects as ways to componentize the programs for maximum reuse. Also, that they are objects with a layout and behavior.

Now we are getting close to the heart of the matter. In any of these objects we have a layout to be designed.

And, besides being able to use third-party controls that already come with design—User Controls— in that layout we can use Stencils to compose controls. But adding design to a Stencil is the equivalent to adding design to any set of controls in a layout.

So now we're getting to where we wanted: regardless of where the GeneXus controls are located, how do we add design features to them?

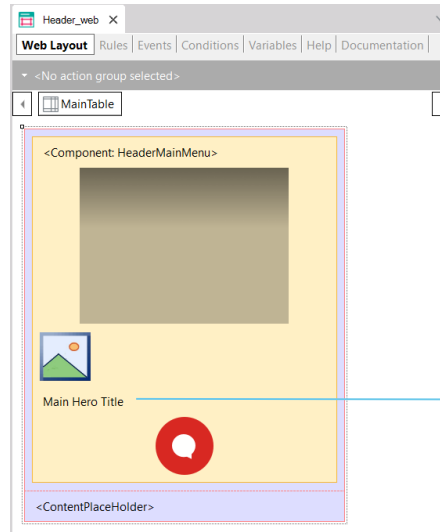
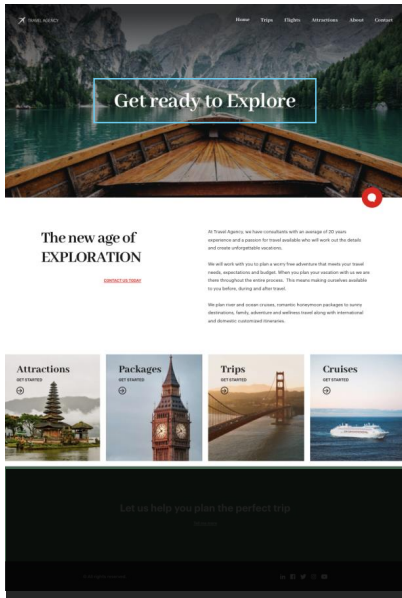
Design System



On the left is the screen as we want to see it at runtime. On the right is the Web Panel Home. It is loaded inside the displayed Master Page, which in turn uses the Web Component. In addition, besides having common controls, the Web Panel uses a Stencil that also has a layout.

The way to add design to each one of the controls of these layouts is using what we know as classes. Each control is associated with one or more classes, which determine how it is displayed on the screen.

Design System



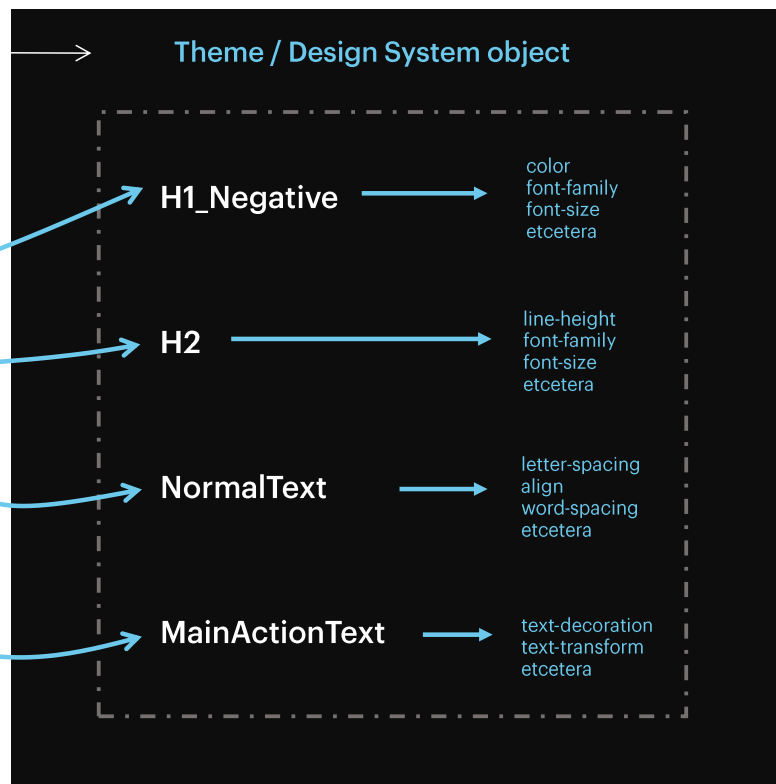
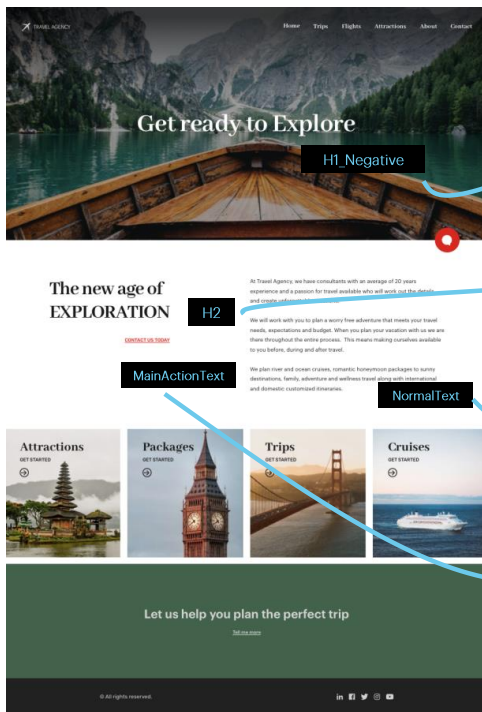
H1_Negative

color: white;
font-family: AbhayaLibre-Bold;
font-Size: 95px;

For example, if we look at the title “Get ready to Explore” corresponding to the text block we see here... how do we set it to be a white title, to have that font family, and that size?

The answer is: by associating it with a class, which is named as we want and is semantically meaningful, for example, H1_Negative, to indicate headings on a dark background. Also, indicating the desired design features at the class level.

The class can be reused in many other controls, and that is the advantage of its design features being independent of the control.



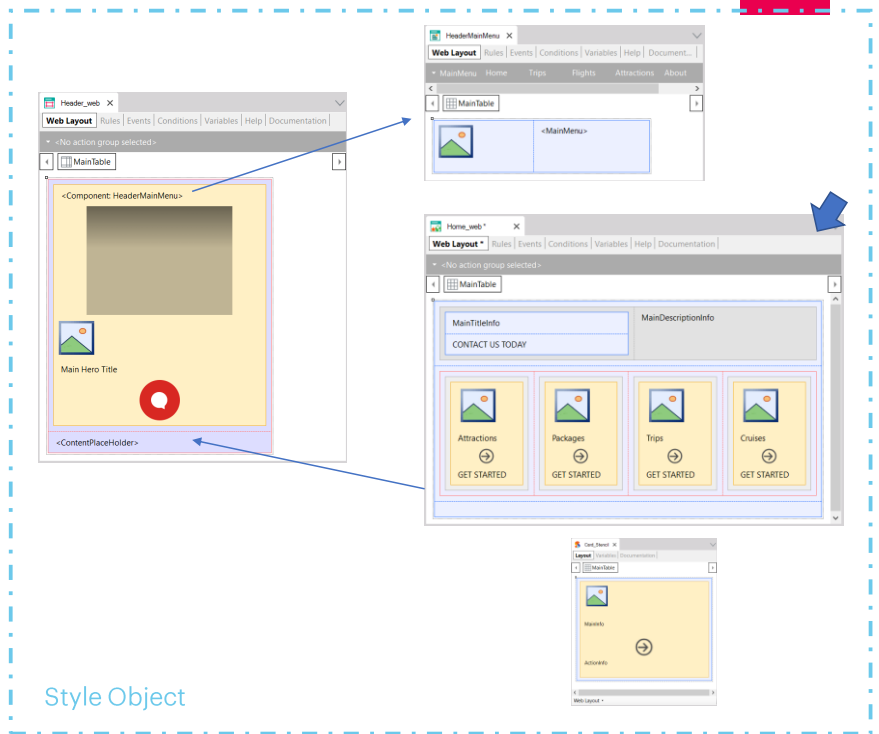
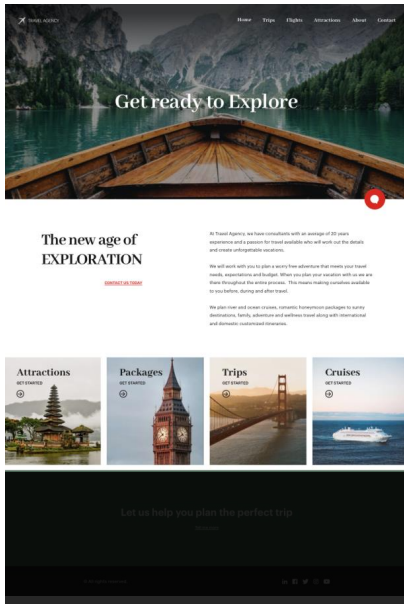
So, the idea is to associate all UI controls or elements in the objects—for example, these 4 texts—with the style classes identified as significant. Then it will only be necessary to define the design properties of each of these classes, in an object that centralizes them.

Until GeneXus 17, this object was the Theme, but from then on it can also be its evolution, the Design System object.

Therefore, the page, which has class names for each control, will only need to know which Design System object or Theme it has to go to in order to retrieve the design properties of each one of those classes.

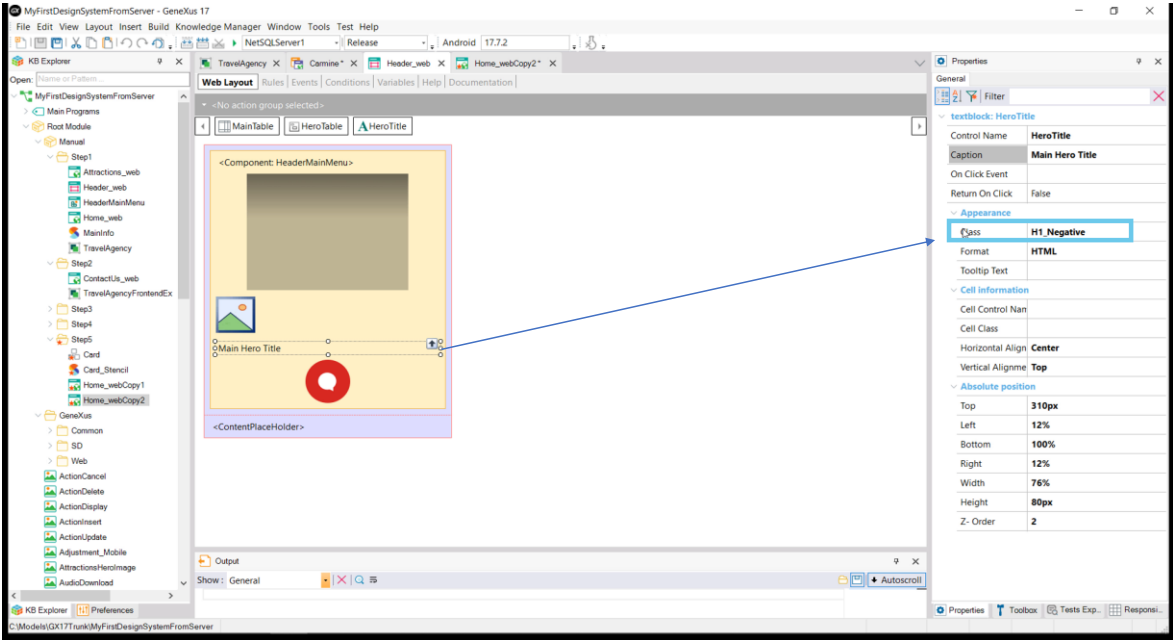
And that is the way GeneXus separates the structure and content of the pages from their design.

Design System

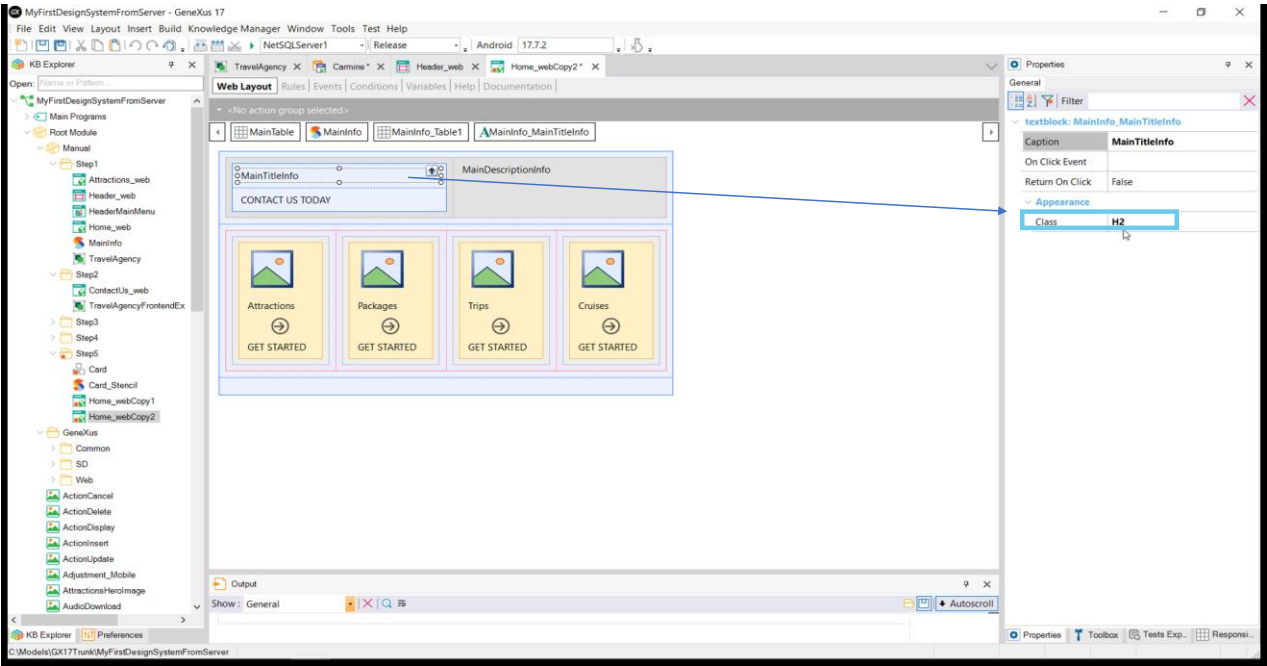


All these objects that implement the page we will see in the browser have an associated style object: either the Theme object or the new Design System object. In this Theme or Design System object, the classes are declared with their specific properties that give them the style they will be displayed with on the screen.

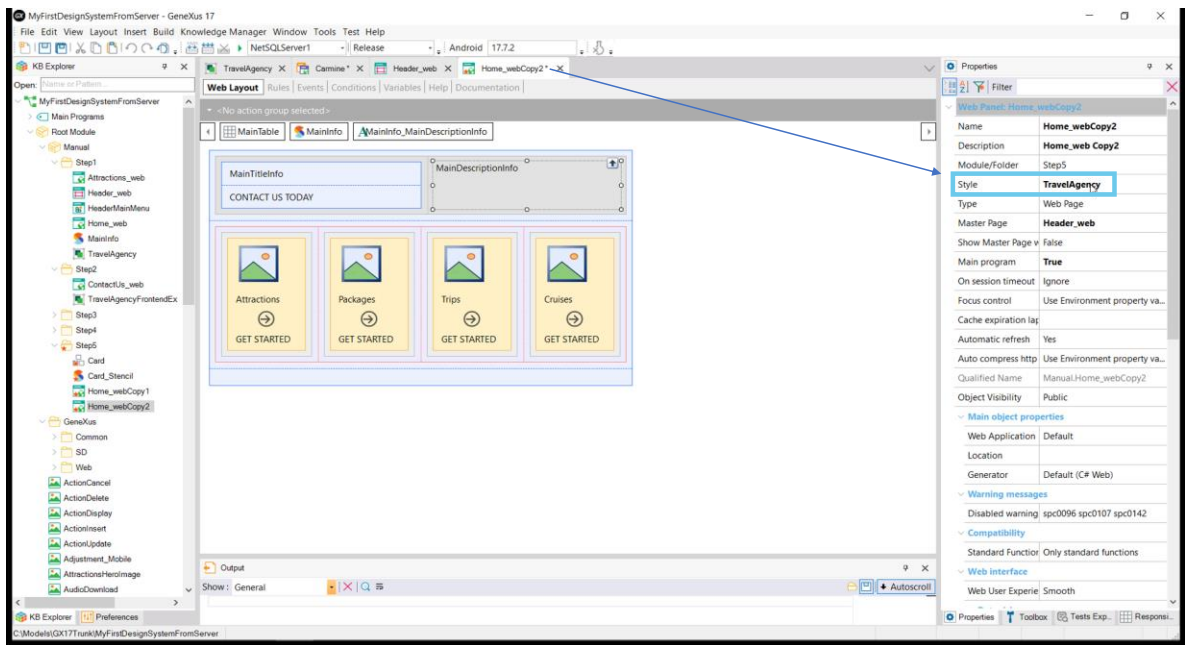
Design System



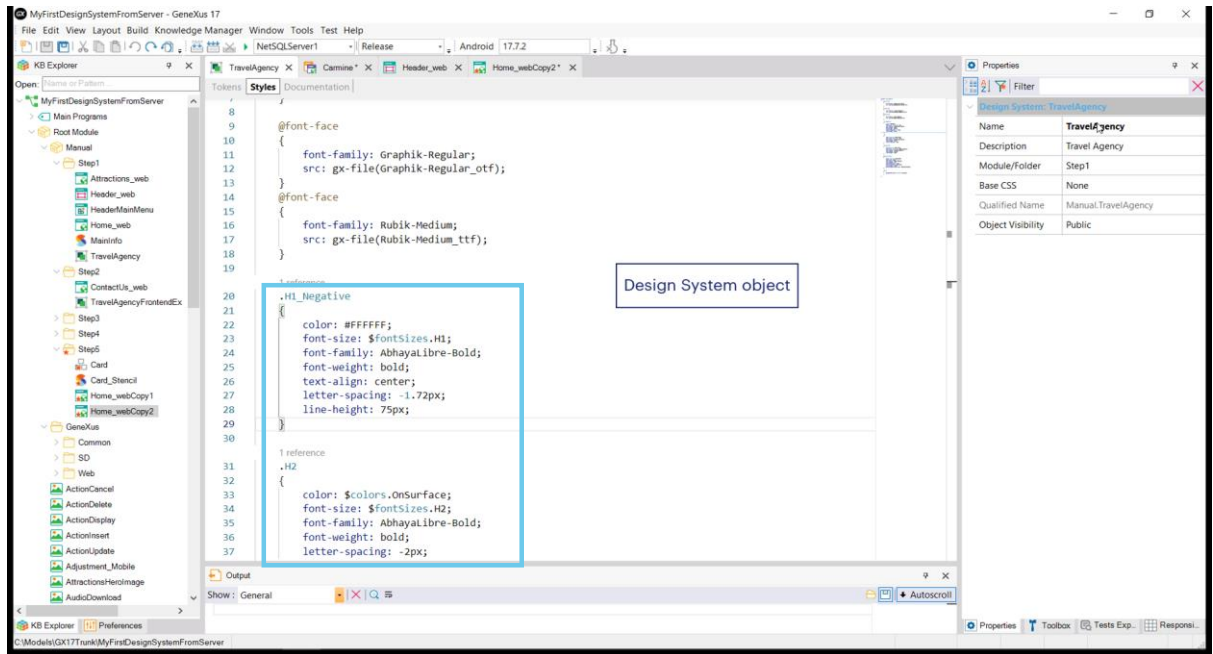
Design System



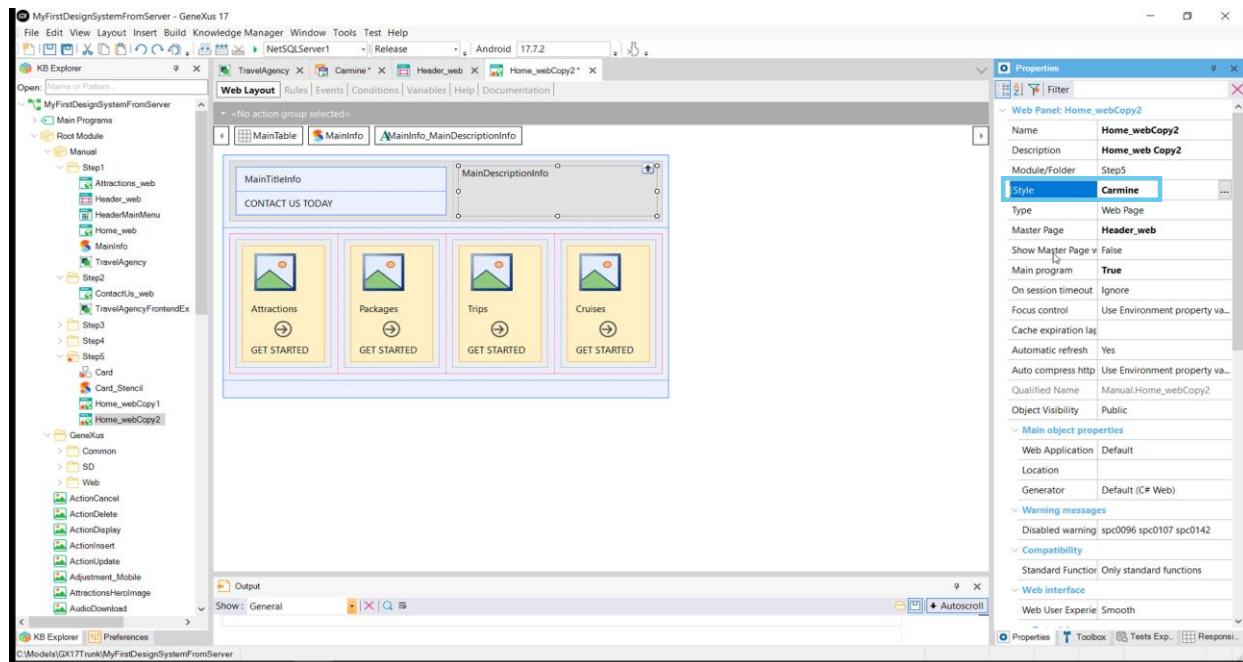
Design System



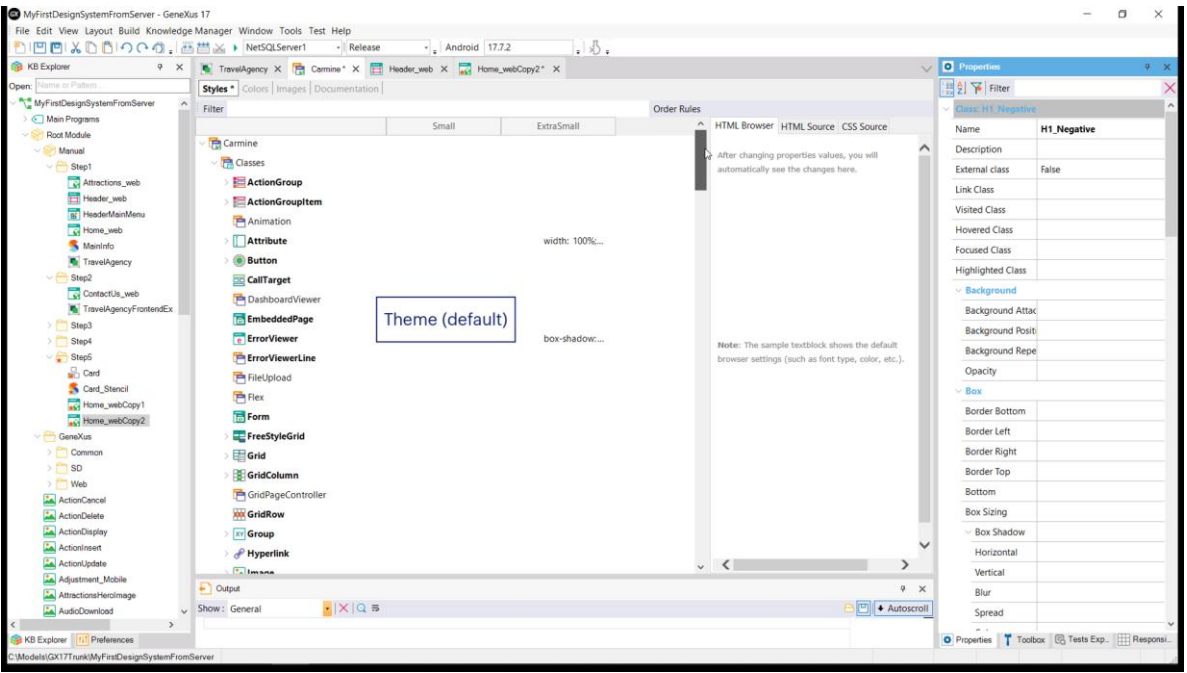
Design System



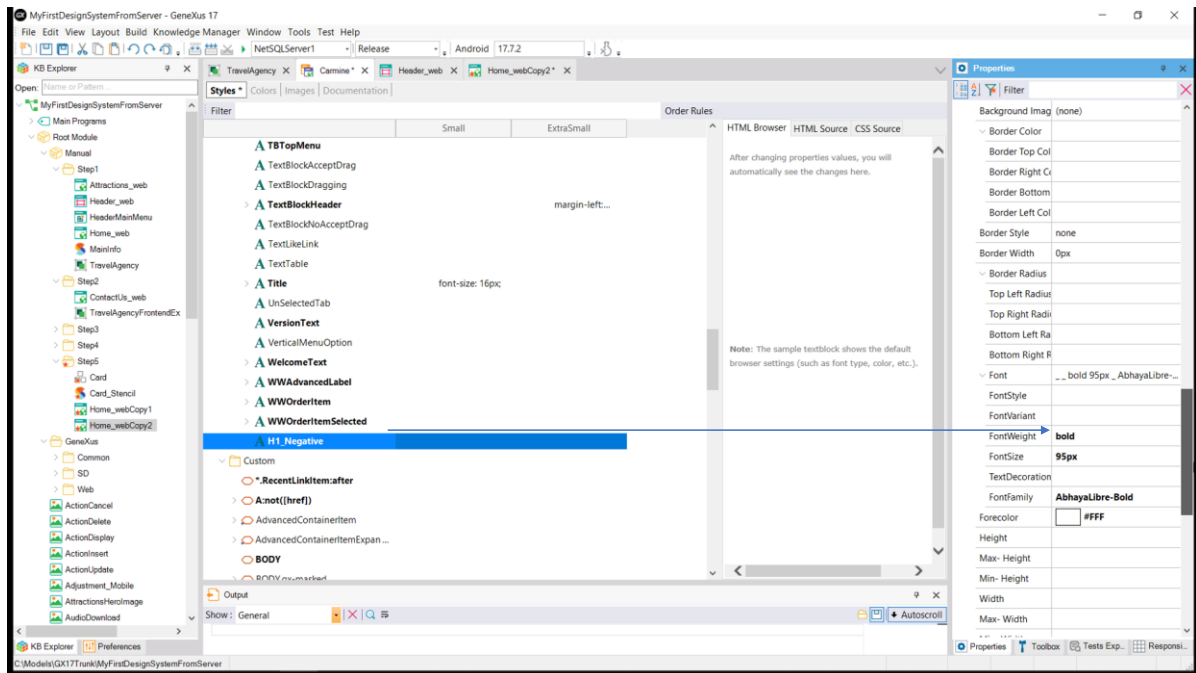
Design System

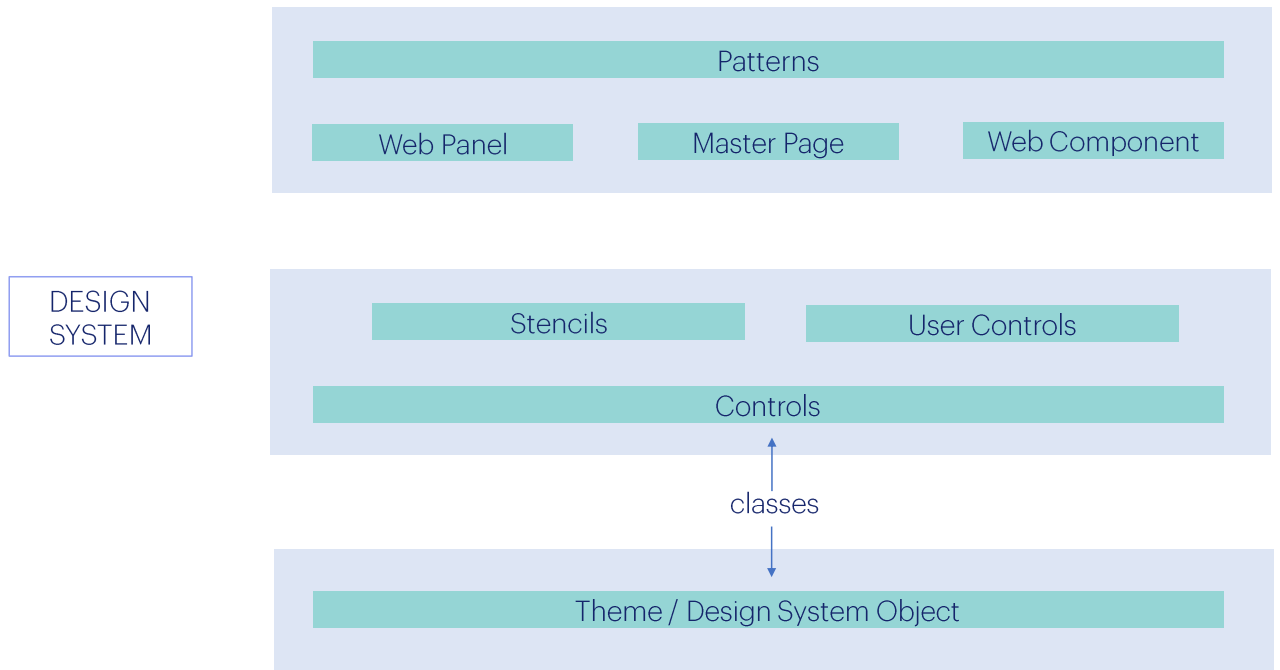


Design System



Design System





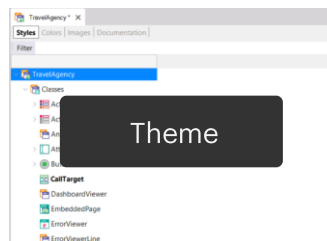
And this is how the controls are designed: through classes whose properties are defined in the Theme or Design System object.

So far, we have only discussed how to develop web applications. However, almost the same will apply to native mobile applications, or even Angular applications.

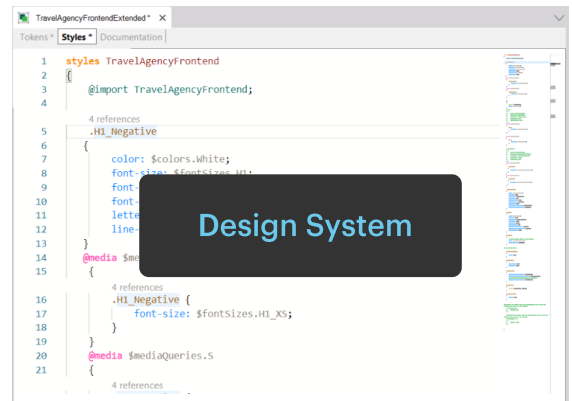
At the time this video was made, the latest upgrade of GeneXus 17 was number 7. There the default style object is still the Theme, but in future upgrades it will be changed to the Design System Object.

All this will become a reality and fully come to life when Unanimó—a complete Design System with a new design for the Pattern and transactions too—is released.

DESIGN
SYSTEM



Before



Now

Wiki: Design Systems (<https://wiki.genexus.com/commwiki/servlet/wiki?40108,Toc%3Design+Systems>)

There is a lot more to say and show, but here we simply wanted to present a complete overview of how the Design System of any application is modeled in GeneXus, at this turning point when we are changing from the use of the Theme object to the new Design System Object.

There is already enough material (in videos and wiki articles) for you to learn more about all this, whenever you want.

*GeneXus*TM

training.genexus.com
wiki.genexus.com