Practice course for upgrade to

GeneXus™ 16

PART 1

April 2020

Copyright © GeneXus S.A. 1988-2020.

All rights reserved. This document may not be reproduced by any means without the express permission of GeneXus S.A. The information contained herein is intended for personal use only.

Registered Trademarks: GeneXus is trademark or registered trademark of GeneXus S.A. All other trademarks mentioned herein are the property of their respective owners.



CONTENTS

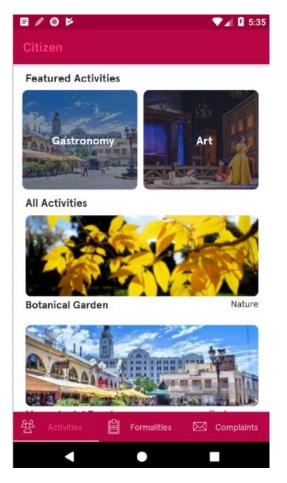
CONTENTS	2
OBJECTIVE	
GETTING STARTED	4
BECOMING FAMILIAR WITH THE KB	4
STENCILS	6
Justification for using stencils	6
Solution:	
Creating and using a stencil	
Possible solution:	
SMART GRID	
Solution:	
FLEX GRID	
Solution:	
BASE STYLE	
USER CONTROL	20
Solution:	
Adding an action to the user control	
KB SOLUTION	



OBJECTIVE

In order to achieve the objective of developing multi-experience and omni-channel applications while enhancing our integration with the design team, this practice session will deal with the topics of Design Systems and user experience related to the ways in which we can make repetitive information to flow more smoothly and in a manner more adaptable in the UI, all of which relates to the new grid types.

We will be working on an application made upon a simplification of an app for a city's municipality that has a Web frontend and an SD frontend for the citizens of that city to use a user identifier for filing claims (such as fallen trees, traffic lights out of order, wrongly parked vehicles and so on), as well as other procedures like obtaining a driver's license, making arrangements for financing amounts due, installing public advertisements, etc.). With the app, they may schedule a date for making their claims or procedures with the city's personnel. The frontend also shows the various cultural activities that the city has to offer:



There is also a web backoffice for some employees to manage data and check statistics.



GETTING STARTED

We will be using the **GeneXus 16** version and the **Android SDK** emulator. The instructions of these practice exercises have been updated according to upgrade 9 of GeneXus 16. If you do them with a later version, remember that there may be differences between what is shown here and what you see.

Unless the instructor tells you that the KB has already been created in your computer, in GeneXus do the following: File/New Knowledge Base from GeneXus Server, selecting Server KB URL: http://samples.genexusserver.com/v16/.

ervers:	Knowledge Bases:	Search by pattern here	H	Knowledge Base	Info:
Open	ApproveProductOrder		^		Citizen_GX16UpdC
Samples	ApprovePurchaseOrder				Created by GXTechnical\sjuare
Sandbox	CISample				September 05, 2018
Add New Server	Citizen_GX16UpdCourse				
	ConsumingGXMServices				No description available
	🔜 DeepLink U6				
	🔂 DemoGX16		L]
	📦 DemoXev3U3SAPMMSD	WebSD			
	📦 EventDay				
	🗹 EventDay15				
	EventDayForSAPSystems				
	C Gene Xus 15AnalystCourse	PracticalExercises			
	GeneXus15JuniorCourseF	PracticalExercises			
	🙀 GenexusMeliSDK		¥		
Name: Citizen_GX16Up	pdCourse				

In Select Server KB, select the name "Citizen_GX16UpdCourse":

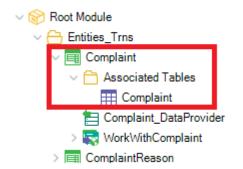
And select the trunk version.

Do a Rebuild All and Create. We will be doing local prototyping and not in the cloud. Make sure that the names of server and database are already set up, or set them up with the instance of the computer's SQLServer and DB name as you wish. Ask your instructor.

BECOMING FAMILIAR WITH THE KB

The entities that model the application's reality are in the Entities_Trns folder. Note that now, tables are shown under each transaction in the KB Explorer.





Also note that the Data Providers have been defined to populate the tables associated with the transactions (linking the Data Provider property of the transaction). They will be invoked automatically upon the first execution.

There are also two folders, FrontendWeb and FrontendSD, that implement the two frontends, with the main objects of the app that the citizens will use.

Web panel Home inside GeneXus/Web contains the backoffice's main object (it is default) that will be used by the city's staff for managing information and the site. It will be deployed as the outcome of the F5, which in turn will display the following error:

된 Oi	utput				
Show :	Build	$\cdot \mid \times \mid$ Find:			
E	====== Executi xecution Success alculating Mobile ======= Androic	Targets for	SmartDevices	•	Devices)
A	rror: A startup o ndroid Execution un Developer Menu	Failed	e selected		
<					

This is due to the fact that we have not set up a Startup object, because we will be moving between Web and SD frontend and backoffice. We can decide to specify the Home of the backoffice as Startup Object, or not, because we will nevertheless have to do Run on the main object we want execute every time (CitizenMenu for SD frontend, and CitizenWeb for Web frontend).

The Queries folder includes queries that will be used in a different practice session.

When you execute the Home web panel:



🗅 Home 🛛 🗙	+	- 🗆 ×
← → C (i) localhost/Citizen_	GX16UpdCourse6.NetEnvironment/home.aspx	☆ ╗ ○ 🕸 :
Citizen Service		
		by GeneXus
Recents		
Complaints Complaint Reasons Cultural Activities Formalities Formality Categories Reservations Traffic Signals Users		

You will see the entities' WorkWiths with the data loaded. Leave that open.

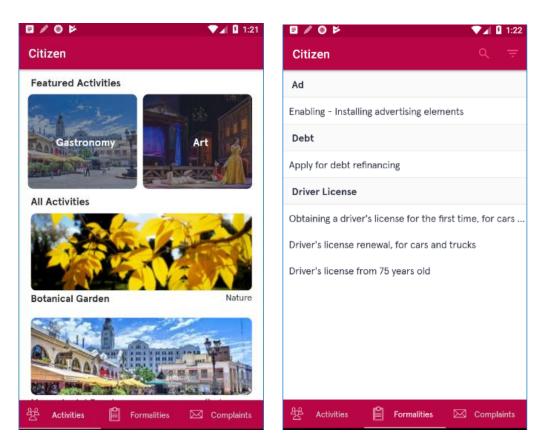
STENCILS

Execute the main object of the SD app (CitizenMenu object in the FrontendSD folder).

JUSTIFICATION FOR USING STENCILS

The first screen you will see shows the cultural activities (Activities). And in the menu's second tab underneath you will see the different claims and procedures available to citizens (installing public advertisements, arrangements for the financing of amounts due, obtaining a driver's license):





When the user selects a procedure, such as an advertisement authorization, the user is shown general information on the procedure and is allowed to book a reservation where he will be carrying out that procedure. When the user presses "BOOKING" he will be asked for the user ID:

E 🖉 O 🎽 🔍 🔽 1:27	
← Ad	← Ad
General Information	General Information
Enabling - Installing advertising elements \$300	Epobling - Installing advertising elemente \$30
	Please, insert your user ID:
Available hours from 02:00 PM to 05:00 PM	User Identification
Requeriments	R
You must submit an express statement from the technician (Architect or Engineer). The form must be complete fields, with all the corresponding signatures (owner, representative of installation company, representative of advertising company, representative of the trade, acting technician) and the respective professional stamps. Plans of location, plant, cut and	CONFIRM
facade of the proposal, respectively bounded.	
Address & Contact	
18 de Julio Av 1360, Montevideo Department 🛛 🍳	
9876541	9.
BOOKING	BOOKING

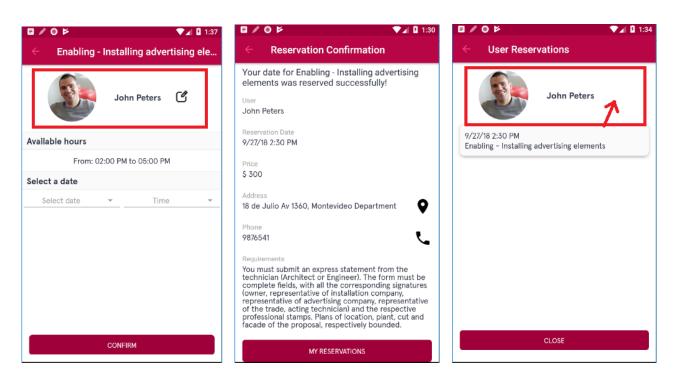


🗅 Users	× +					-	٥
- → C (0	localhost/Citizen_GX16UpdCou	rse6.NetEnvironment/wwuser.aspx				☆ ୭フ	0 🥸
Citize	en Service					by Ger	neXuš
Recents	Users						
ι	Jsers	Q Name				+1	NSERT
	Identification	Name	Photo	Age			
	20576	Ann Bert	.	43	UPDATE	DELETE	
	11182	Brian Goals		77	UPDATE	DELETE	
	89680	Cecilia Fernández	2	43	UPDATE	DELETE	
	15587	Cecilia Lemos		16	UPDATE	DELETE	
	81180	Eleonor Johnson		28	UPDATE	DELETE	
	49128	Jessica Loyarte		30	UPDATE	DELETE	
	12345	John Peters		45	UPDATE	DELETE	
	11245	Luciano Silveira		42	UPDATE	DELETE	

We recommend that you open the Backoffice to view the users registered and their IDs:

When you enter one and confirm (such as ID 12345), you get a screen showing a photo and the name of the user entered, with the possibility of booking a day and time, within the office hours allowed for the procedure, so as to book a reservation for carrying out the procedure. When you do this, if it all works out fine, you will get another screen showing all the information regarding that procedure, and you will be offered the option of viewing all the bookings you have made so far:

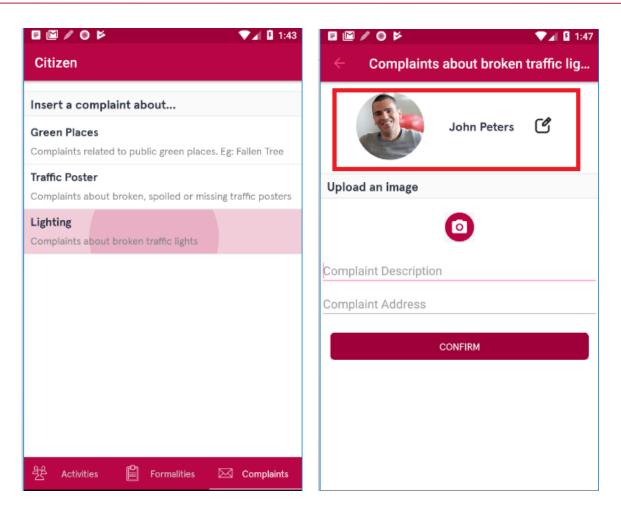




In this last screen, you will also view the user's photo and name, with the same design as previously, but with the possibility of changing users now hidden.

By going to the Complaints menu option and selecting the claim type, the user will be able to enter one of that type into the system:





Note that, on this screen, UserComplaint, the user information is also shown in the same manner as in the procedure.

To modify the manner in which the photo is shown (on the three screens on which it appears) so that it will not appear with a round shape anymore, as shown here:



◼◙∕⊙⊳				2:09		
← Driver's license renewal, for cars a						
	Johr	Peters	C			
Available hours						
From:	03:00 PM t	o 05:00 PN	4			
Select a date						
Select date	.	Time	3	Ŧ		
	CONFIR	м				

How could this be implemented?

SOLUTION:

Open the UserProfile stencil and see that the photo is associated with the ImageUserProfile class of theme CarmineSDCourse. Change the Border Radius class and leave it empty.

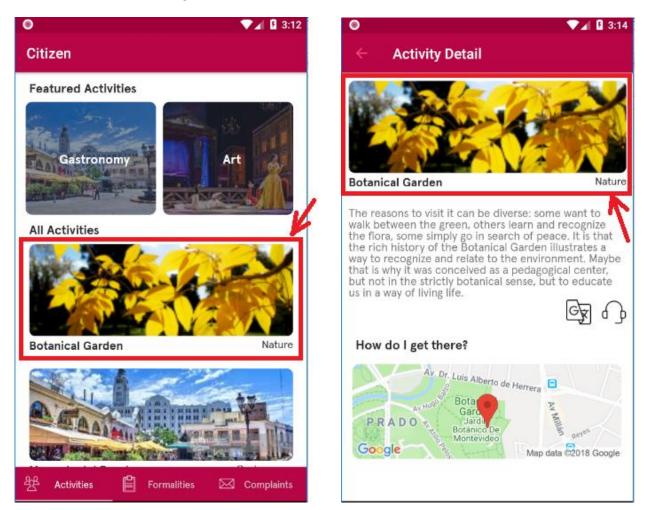
• Properties	7	×
Filter		×
V Class: ImageUserPr	ofile	
Name	ImageUserProfile	
Border Radius	Julip	
Scale Type	Fill Keeping Aspect Ratio	
Placeholder Image	userAvatarPlaceholder_3x	

Execute CitizenMenu and note the change on the three screens.



CREATING AND USING A STENCIL

Now we want the screen that shows the cultural activities (Activities) to show, on top, the outstanding activities that are highlighted, with the remaining activities underneath. When you select an activity in either grid, you will view a screen with a detail:



Note that the activity, as well as its name and category, are shown in a manner identical as in the grid that shows All Activities in the first panel and in the second panel. To avoid repeated designs we use stencils.

Implement a stencil to show this information and use it in both panels: Activities and ActivityDetail.

POSSIBLE SOLUTION:



Create the stencil called CulturalActivityHeader (you can do this manually, or you can click on the table that already contains the photo and both attributes in the Activities pane, and right-click to select Wrap as new stencil):

😞 CulturalActivityHeader * 🗙 🛅 CarmineSDCourse * 🗙	\sim	Properties	Р ×
Layout * Variables Documentation		General Class	
MainTable CulturalActivityPhoto		📲 🧎 🌾 🛛 Filter	×
	^/	✓ Table: MainTable	
		Control Name	MainTable
		~ Appearance	
		Columns Style	70%;30%
		Rows Style	130dip;pd
		Width	100%
		Height	100%
		Auto Grow	True
&CulturalActivityName &CulturalActivityCategory ~	-	Class	TableCulturalActivityList ~
		Background	(none)
		Visible	True
		Invisible Mode	Keep Space
		Enabled	True
	×		
SD 🗸			

If you did it manually, set the properties for the variable with the image as follows:

- Label Position: None
- Class: ImageCulturalActivityList
- Col Span: 2

Set up the properties for the variable that shows the name on the left, as follows:

- Label Position: None
- Class: AttributeActivityNameList

Set up the properties for the variable that shows the category on the right, as follows:

- Label Position: None
- Class: AttributeActivityCategoryList
- Horizontal Alignment: Right

In the ActivityDetail panel, and in the table where you would insert the CulturalActivityPhoto, CulturalActivityName and CulturalActivityCategory attributes, now empty, search for the Stencil you have just created and insert it.



	le 🗙 🛐 CarmineSDC les Events Conditions			S Cult	uralActivityHeader	×	Activity	Detail 🗙		<>マ
 Application 	on Bar									
. → Main	Table Table2									→
How do	ActivityDescription ActivityDescription I get there? Im, Default Orientations	 Insert Butt Delete Convert to Convert to Wrap as N Swap table Go To Even Insert Row Insert Column 	SUPP Canvas Flex ew Stencil e with Stencil nt Before After umn Before umn After							
Show : Gener	ral 🗾 🔽								60	Autoscroll
Layout Rules	X The conditions Var Events Conditions Var elect Objects Filters Pattern: Category: *ALL Modified:					stencil	×	<		Properties eral Class ×
Cultura How do Any Platfo Output Show : Gene	Root Module CulturalActivi 30/3/2020 18:50 Cultural	ityName &CulturalActivit	yCategory		Root Module Use 8/8/2019 11:41	arProfile User Pro	&UserNar Stille		Ċ	
	< 1 selected object(s).							ОК	2 matching)) object(s). ancel

Note:



• You must replace the variables with the CulturalActivityPhoto, CulturalActivityName and CulturalActivityCategory attributes.

If you created the stencil manually, in the Activities panel go to the grid table that shows all the activities, right-click and note that, when you select "Swap table with stencil":

😭 CarmineSDCourse 🗙 🚮 Activities* 🗙 🕵 Cultura	lActiv	ityHeader 🗙 <u>त</u> ActivityDetail 🗙	<
Layout * Rules Events Conditions Variables Docum	nenta	ation	
✓ Application Bar			
MainTable Table1 Grid1 Grid1	Table	Table2	
CulturalActivityCategory			
All Activities			
GRID			
	=11	Insert Attribute/Variable	
		Insert Button	
	X	Delete SUPR	
CulturalActivityName CulturalActivityCategor	5	Convert to Canvas Convert to Elex	
	5	Wrap as New Stencil	
	5	Swap table with Stencil	
Any Platform, Default Orientations - 🕈 Add Layout 😢		Go To Event	
E Output	••	Insert Row Before	
Show: Build · X Q ⇒	••	Insert Row After	
BUILD SUCCESSFUL in 11s	4	Insert Column Before	
25 actionable tasks: 3 executed, 22 up-to-date	1	Insert Column After	
Copying C:\Models\GX16Trunku9\Citizen_GX16UpdC	1	Delete Row	id\(
Success: Android Compilation	8	Delete Column	

The table will be replaced with the stencil, keeping the attributes (they are not replaced with variables.

Test at runtime (Run on the CitizenMenu object).

SMART GRID

The cultural activities panel shows, on top, a grid with the activities marked and highlighted (see CulturalActivityFeatured attribute of transaction CulturalActivity). Those activities should be viewed in a carousel formal, as shown in the image:





How is this implemented?

SOLUTION:

Featured Activities		O Properties		7
	1	General Class		
GRID GRID		🔡 🛃 🌾 Filter		
		 Grid: Grid2 		
		Control Name	Grid2	
	1	Collection		
		Default Action	'ViewDetail'	
		Selection Type	Platform Default	
CulturalActivityCategory V		Enable Multiple Selection	False	
0		Pull To Refresh	False	
All Activities		Inverse Loading	False	
SRID 👻	<u>۱</u>	Default Selected Item L	i (none)	
[]	- L	✓ Control Info		
•		Control Type	SD Smart Grid	~
		Auto Grow	False	
		Scroll Direction	Horizontal	
		Snap To Grid	False	
CulturalActivityName CulturalActivityCategory V			Single	

Change the Canvas Width as well, so that it occupies 50% instead of 100%:

16



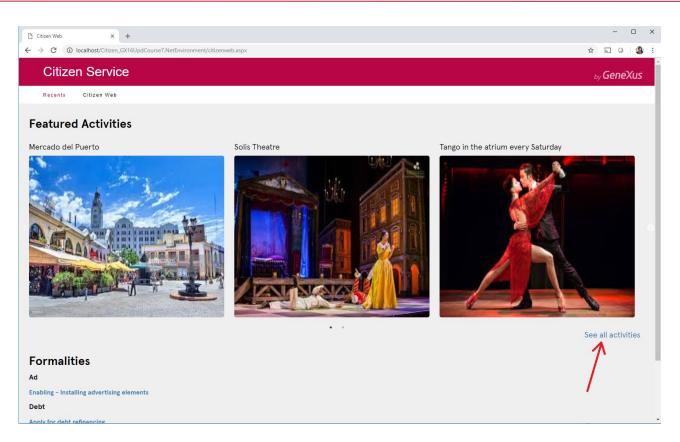
🔂 Activities * 🗙	\sim	• Properties	7 ×
Layout * Rules Events Conditions Variables Documentation		General Class	
- Application Bar		📜 🛃 🌾 Filter	×
✓ MainTable		✓ Canvas: Grid2Canvas	
		Control Name	Grid2Canvas
		Layout Name	
Featured Activities		✓ Appearance	
o GRID		Width	50%
		Height	100%
		Class	TableFeaturedActivities
		Background	(none)
		Visible	True
		Invisible Mode	Keep Space
CulturalActivityCategory ~		Enabled	True
0		∨ Scroll Behavior	
All Activities		Scroll Factor	1
		Zoom Factor	0
GRID +		Scroll Attachment	Parent
		✓ Layout Behavior	
		Expand Bounds	Background Only
		Expand Bounds Dire	Top, Left, Bottom, Right
Any Platform, Default Orientations - 🕂 Add Layout 😵 Delete Layout			

At runtime, note the difference with the horizontal grid.

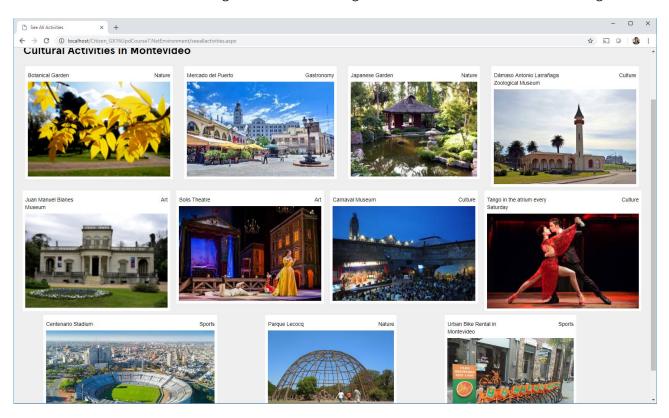
FLEX GRID

Notice that, in the main panel of the web frontend (called CitizenWeb), apart from viewing the highlighted cultural activities, there is an option called "See all Activities":





In this other screen we want the grid to allow showing them in a variable manner, according to size:



How is this implemented?



SOLUTION:				
SeeAllActivities * ×	\sim	O Properties	q	×
Web Form * Rules Events Conditions Variables Help Documentation		General Class		
- Actions Insert		📜 🛃 🌾 Filter		X
MainTable Grid1		Cell Spacing	2	^
	Ц	✓ Appearance		
Cultural Activities in Montevideo		Class	FreeStyleGrid	
		Rows	<unlimited></unlimited>	
		Custom Render	FlexGrid	
		Empty Grid Text		
CulturalActivityName CulturalActivityCategory ~		> Collapsing		
		✓ FlexGrid		
		Flex Direction	Row	
		Flex Wrap	Wrap	
		Justify Content	Space Around ~	~
0 0		Align Items	Stretch	
		Align Content	Stretch	

Note the differences with the Horizontal Grid you had before.

BASE STYLE

When you edit the KB's files you will find the following:

Files ×	\sim
Add New File 🔀 Delete Filter	
Files	- 1
ApercuPro-Regular	
ApercuPro-Bold	
SemanticUI_gxlibrary	

You will also find a new "Base Style" property in the objects:

- Theme
- User Control

This property enables you to assign a base style to these objects.



The property will take the values of the KB's files with extension "gxlibrary" (they are a zip containing all files css, js, assets, etc. that are provided by the designers or the Design System used).

At the theme level you will find the following property:

V Customization		
> 🔁 Localization	Properties	×
✓ ➡ Themes	📜 🛃 🌾 Filter	×
🖂 😽 Carmine	Theme: CarmineW	eb
CarmineRTL		
> 🔽 🛅 CarmineSD	Name	CarmineWeb
🔤 📑 Flat	Description	Carmine Web
GeneXusXEv2	Base Style	Bootstrap v3 🗸 🗸
C SimpleAndroid	Tune	
C SimpleIOS	Туре	None
CarmineSDCourse	Dark Theme	Bootstrap v3
📑 CarmineWeb	Color Palette	Bootstrap v3 RTL
> Colors	Oualified Name	SemanticUI
🔼 Images		Jennandicon
Files		

And at the object level we will see it as soon as we create a User control.

USER CONTROL

If we now select a procedure from the main page of the web frontend:



Citizen Web x +		- 0 X
← → C ① localhost/Citizen_GX16UpdCourse7.NetEnvironment/citizenweb.aspx?		☆ ਙ o ∰ :
Featured Activities		
Tango in the atrium every Saturday	Centenario Stadium	Utable Bike Rental in Montevideu Image: Imag Image: Image: Imag
Formalities		
Ad		
Enabling - Installing advertising elements		
Debt		
Apply for debt refinancing		
Driver License		
Obtaining a driver's license for the first time, for cars and trucks		
Driver's license from 75 years old		

As it was the case with the native mobile app, we will be required to provide a user identifier in order to book a reservation for that procedure:

Formality Reservation Web × +		– 🗆 ×
← → C () localhost/Citizen_GX16UpdCourse7.NetEnvironme	nt/formalityreservationweb.aspx?DriverLicenseRenewal	☆ ╗ ⊘ 🥸 :
Citizen Service		by GeneXus
Recents See All Activities — Citizen Web — Formality	Reservat	
Make a	Reservation	
User Identification	1	
Formality Description	Driver's license renewal, for cars and trucks	
Formality Requirement	Identity card valid and in good condition, and a photocopy of it. Specific Medical Certificate for a driver's license, issued by one of the authorized institutions.	
Formality Price	\$ 400	
Formality Address	18 de Julio Av 1360, Montevideo Department	
From:	То:	
03:00 PM	05:00 PM	
Formality Date Time	/ / 12:00 AM	
Confirm		

Upon entering a valid user ID and exiting the field, we will want to view the user information as follows:



Make a Re	eservation
User Identification	<page-header><page-header><section-header><section-header></section-header></section-header></page-header></page-header>
Formality Description	Driver's license renewal, for cars and trucks
Formality Requirements	Identity card valid and in good condition, and a photocopy of it. Specific Medical Certificate for a driver's license, issued by one of the authorized institutions.
Formality Price	\$ 400
Formality Address	18 de Julio Av 1360, Montevideo Department
From: 03:00 PM	To: 05:00 PM

Note that the Card is loaded onto the component called WCUserInformation loaded on the web form:



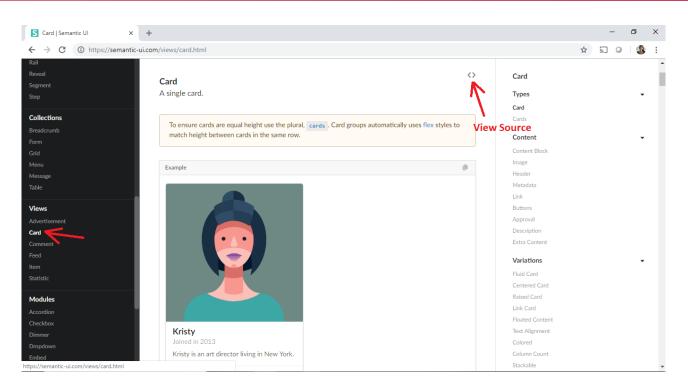
G FormalityReservationWeb × Web Form Rules Events Conditions Variables Help Documentation
<no action="" group="" selected=""></no>
ImainTable
Make a Reservation
User Identification
<component></component>
Formality Description FormalityDescription
Formality Requirements FormalityRequirements
Formality Price FormalityPrice
Formality Address FormalityAddress
From: FormalityBusinessHoursInitial To: FormalityBusinessHoursEnd
Formality Date Time &FormalityDateTime
<errorviewer: errorviewer1=""></errorviewer:>
Confirm

How would you implement that component (WCUserInformation) that is empty in the KB?

SOLUTION:

It is a SemanticUI Card control. If you go to the SemanticUI site and search for Card, you will find different types of these controls. Select, for example, the simplest Card and view the html, in order to copy it:



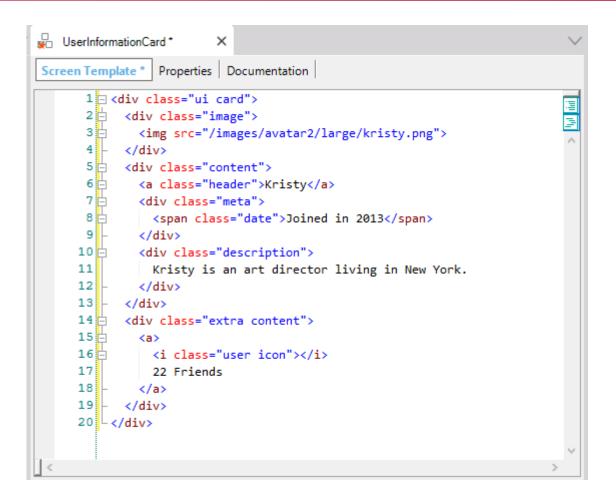


It is written below so you can copy it later and save time during the session:

```
<div class="ui card">
  <div class="image">
    <img src="/images/avatar2/large/kristy.png">
  </div>
  <div class="content">
    <a class="header">Kristy</a>
    <div class="meta">
      <span class="date">Joined in 2013</span>
    </div>
    <div class="description">
      Kristy is an art director living in New York.
    </div>
  </div>
  <div class="extra content">
    <a>
      <i class="user icon"></i>
      22 Friends
    </a>
  </div>
</div>
```

Create a User Control object in GeneXus and copy the previous html in your "Screen Template":





Parameterize the properties according to what you wish to view. In our case, it will be as follows:

UserInformationCard* X	\sim	Properties		무	×
Screen Template * Properties Documentation		📜 🛃 🌾 Filter			\times
1 div class="ui card card-extra-width">		V User Control: Us	serInformationCard		
<pre>2 <div class="image"> 3 </div></pre>		Name	UserInformationCard		
4 -		Description	User Information Card		
5		Module/Folder	Root Module		
7 □ <div class="meta"></div>		Is Control Type	False		
<pre>8 </pre> {{UserBirthDay}} 9 -		References			
10 div class="description">{{UserAddress}}		Base Control Tvp	e None		
11 div class="description">{{UserEMail}} 12 -		Base Style	SemanticUI		\sim
13 L		Qualified Name	UserInformationCard		
		Object Visibility	Public		
<	>				

Below is the previous code written down for you to copy:

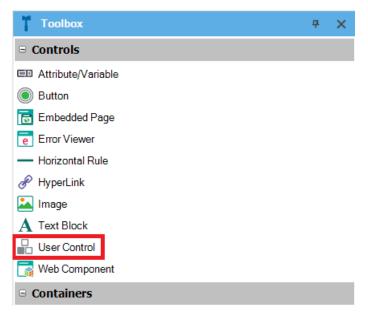
```
<div class="ui card card-extra-width">
```



```
<div class="image">
    <img src="{{UserPhoto}}">
    </div>
</div>
<div class="content">
    <a class="content">
     <a class="header">{{UserName}}</a>
     <div class="meta">
          <span class="date">{{UserBirthDay}}</span>
          </div>
          <div class="description">{{UserAddress}}</div>
          <div class="description">{{UserEMail}}</div>
          </div>
          <div class="description">{{UserEMail}}</div>
          </div>
        </div>
        </div>
        </div>
        </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div>
    </div
    </di>
```

Verify that the value defined for the Base Style property of the User Control object is "SemanticUI". Otherwise, select that value yourself.

You will have to insert the user control you just created in the WCUserInformation component. In the Toolbox, you will find:



Drag it to the Web form and select the User Control you created before:



B WCUserInformation ×	\sim
Web Form Rules Events Conditions Variables Help Documentation	
✓ <no action="" group="" selected=""></no>	
MainTable	+
<userinformationcard: userinformationcard1=""></userinformationcard:>	

Then initialize the properties of the User Control based on the attributes:

🐻 WCUs	erInformation * ×	\sim
Web Form	Rules Events* Conditions Variables Help Documentation	
Events	~	
1 2 3 4 5 6 7	UserInformationCard1.UserName = UserName UserInformationCard1.UserBirthday = "Born: " + UserBirthDate.ToString() UserInformationCard1.UserAddress = "Address: " + UserAddress UserInformationCard1.UserEMail = "E-Mail: " + UserEmail	> ¹ 14 hi
8 9 10	- EndEvent	×

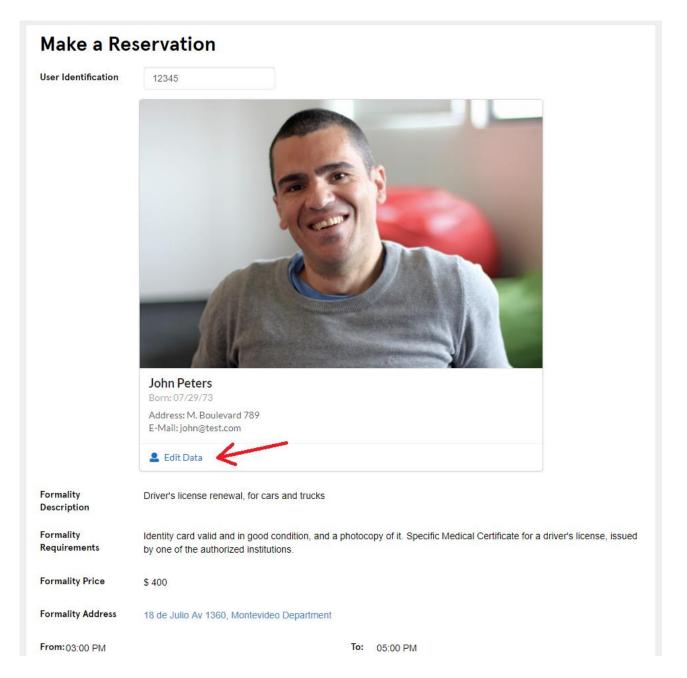
Note: To verify that the Base Library (in this case SemanticUI) was correctly loaded, check the web directory of the KB (Tools \rightarrow Explore Target Environment Directory) and see if the "SemanticUI" folder exists. If not, we suggest that you do a Rebuild of the CitizenWeb object.

Now execute.

ADDING AN ACTION TO THE USER CONTROL

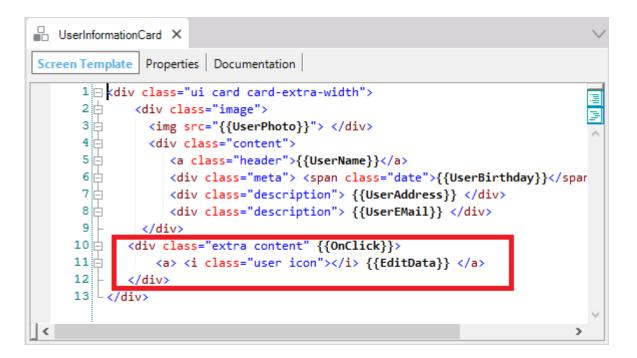


There is the option to allow that the user information be edited from the Card.



It was not exactly like this in the source html of the SemanticUI control. You may customize its User Control object by adding:





And then, in the Web component:

🔐 UserInformationCard X 👼 WCUserInformation* X	\sim
Web Form Rules Events* Conditions Variables Help Documentation	
Load v	
<pre>1 Event Load 2 UserInformationCard1.UserPhoto = UserPhoto.ImageURI 3 UserInformationCard1.UserName = "Born: " + UserBirthDate.ToString() 4 UserInformationCard1.UserAddress = "Address: " + UserAddress 5 UserInformationCard1.UserEMail = "E-Mail: " + UserEmail 6 UserInformationCard1.EditData = "Edit Data" 7 Endevent</pre>	- ¹ 14 ini
7 L Endevent 8	>

Test it at runtime.

And now the event must be programmed. To do that:



👷 UserInfor	rmationCard 🗙 👼 WCUserInformation 🗙	\sim
Web Form	Rules Events Conditions Variables Help Documentation	
UserInformationCard1.OnClick ~		
2 3 4 5 6 7	<pre>Event Load UserInformationCard1.UserPhoto = UserPhoto.ImageURI UserInformationCard1.UserName = "Born: " + UserBirthDate.ToString() UserInformationCard1.UserAddress = "Address: " + UserAddress UserInformationCard1.UserEMail = "E-Mail: " + UserEmail UserInformationCard1.EditData = "Edit Data" Endevent</pre>	- ¹ 1 <u>0</u> 1 <u>11</u>
8 91 10 11	Event UserInformationCard1.OnClick User(TrnMode.Update, UserIdentification) Endevent	>

Test again.

KB SOLUTION

You may download the solution KB for this practice session from GeneXus Server in order to compare results. It is the version of the KB called CitizenSolutionPartOne.



MONTEVIDEO - URUGUAY CIUDAD DE MÉXICO - MÉXICO MIAMI - USA SÃO PAULO - BRASIL TOKYO - JAPAN

Av. Italia 6201- Edif. Los Pinos, P1 Hegel N° 221, Piso 2, Polanco V Secc. 7300 N Kendall Drive, Suite 470 Rua Samuel Morse 120 Conj. 141 2-27-3, Nishi-Gotanda Shinagawa-ku, Tokyo, 141-0031