Curso práctico de actualización a



PARTE 1

Abril 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.



CONTENIDO

CONTENIDO	2
OBJETIVO	3
COMENCEMOS	4
FAMILIARIZACIÓN CON LA KB	4
STENCILS	6
Justificación del uso de stencils	6
Solución:	
Crear un stencil y usarlo	11
Posible solución:	
SMART GRID	
Solución:	
FLEX GRID	
Solución:	
BASE STYLE	
USER CONTROL	
Solución:	
Agregar una acción al User control	27
KB SOLUCIÓN	



OBJETIVO

Para cumplir el objetivo de desarrollar aplicaciones multi-experiencia y omnichannel, mejorando la integración con el equipo de diseño, en este práctico trataremos los temas Design Systems y experiencia de usuario relacionada con las maneras de lograr que la información repetitiva fluya de manera más continua y adaptable en la UI –todo lo que hace a los nuevos tipos de grid–.

La aplicación sobre la que trabajaremos es una simplificación de una app para la municipalidad de una ciudad, que ofrece un frontend Web y uno SD para que los ciudadanos mediante su identificador de usuario puedan realizar reclamos (por ejemplo por árboles caídos, semáforos que no funcionan, coches mal estacionados, etc.), puedan realizar trámites (por ejemplo para obtener licencia de conducir, refinanciar una deuda con el municipio, instalar elementos de publicidad, etc), reservando un turno para ser atendidos por el personal del municipio. En el frontend se les muestran también las diversas actividades culturales que ofrece la ciudad:



Y también se cuenta con un backoffice web para que ciertos funcionarios de la municipalidad manejen los datos y vean estadísticas.



COMENCEMOS

Utilizaremos la versión **GeneXus 16** y el Emulador de **Android SDK.** La letra de este práctico se ha actualizado de acuerdo al upgrade 9 de GeneXus 16. Si lo realiza con una versión posterior, considere que puede haber cambios entre lo que aquí se le muestra y lo que usted ve.

A menos que los instructores le digan que la KB ya está creada en su máquina, haga en GeneXus: File/New Knwoledge Base from GeneXus Server, eligiendo como Server KB URL: <u>http://samples.genexusserver.com/v16/</u>.

Select Knowledge Base from GeneXus Server X						
Servers:	Knowledge Bases:	Search by pattern here		Knowledge Base I	nfo:	
Open Samples Sandbox Add New Server	ApproveProductOrder ApprovePurchaseOrder ClSample Citizen_GX16UpdCourse ConsumingGXMServices DeepLinkU6 DemoGX16 DemoXev3U3SAPMMSDW EventDay EventDay EventDay EventDay15 ConeXus15AnalystCourseP	ebSD acticalExercises	^		Citizen_GX16UpdCo Created by GXTechnical\sjuarez September 05, 2018 No description available	
Name: Citizen_GX16UpdCo	Genexus IsJuniorCoursePra	icticalExercises	¥		Select Cancel	

En Select Server KB elija la de nombre "Citizen_GX16UpdCourse":

Y seleccione la versión trunk.

Realice un Rebuild All y Create. Prototiparemos localmente y no en la nube. Asegúrese de que ya estén configurados nombre de server, y de base de datos, o configúrelos con la instancia del SQLServer de la máquina y nombre de DB la que usted desee. Consulte al instructor.

FAMILIARIZACIÓN CON LA KB

Bajo el folder Entities_Trns se encuentran las entidades que modelan la realidad de la app. Observe que ahora las tablas se muestran bajo cada transacción en el KB Explorer.





Asimismo observe que se han definido los Data Providers para poblar las tablas asociadas a las transacciones (prendiendo la propiedad Data Provider de la transacción). Serán invocados automáticamente en la primera ejecución.

Luego, se cuenta con dos folders FrontendWeb y FrontendSD que implementan ambos frontends, con los mains de la app que usarán los ciudadanos.

El Web panel Home dentro de GeneXus/Web contiene el main del backoffice (es el default), que será utilizado por los funcionarios del municipio para la administración de la info del sitio. Será el que se le desplegará como resultado del F5, que a su vez desplegará el siguiente error:

€) C	Jutput
Show	Build Find:
	======== Execution started ======== Execution Success Calculating Mobile Targets for SmartDevices (Smart Devices) ========= Android Execution started ========= error: A startup object must be selected
	Android Execution Failed Run Developer Menu Failed

Esto es porque no configuramos Startup object, debido a que vamos a estar alternando entre backoffcie y frontend Web y SD. Puede especificar al Home del backoffice como Startup Object o no. Como desee, ya que de todas maneras deberá hacer Run sobre el main que desee ejecutar en cada oportunidad (CitizenMenu para frontend SD, CitizenWeb para frontend Web).

El folder Queries cuenta con unas consultas que se utilizarán en otro práctico.

Si ejecuta el web panel Home:



🗅 Home X +	- 🗆 ×
← → C ③ localhost/Citizen_GX16UpdCourse6.NetEnvironment/home.aspx	☆ ╗ ⊘ 🅵 :
Citizen Service	
	by GeneXuš
Recents	
Complaints	
Complaint Reasons	
Cultural Activities	
Formalities	
Formality Categories	
Reservations	
Traffic Signals	
Users	

Podrá ver los work withs de las entidades, con los datos cargados. Déjelo abierto.

STENCILS

Ejecute el main de la app SD (objeto CitizenMenu dentro del folder FrontendSD).

JUSTIFICACIÓN DEL USO DE STENCILS

Verá una primera pantalla que muestra las actividades culturales (Activities). Y en el segundo tab del menú que se encuentra debajo podrá ver los trámites que el usuario puede realizar (instalar publicidad, refinanciar una deuda, obtener licencia de conducir):





Si el usuario elige un trámite, por ejemplo, habilitar publicidad, se le muestra la información general del trámite, y se le permite al usuario reservar un turno para ser atendido para realizar ese trámite. Y al presionar el "BOOKING" se le pide el ID de usuario:

E / O Þ 🔽 1:27	
← Ad	← Ad
General Information	General Information
Enabling - Installing advertising elements \$ 300	Enabling - Installing advertising elements \$300
	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	Y te c c c c c c c c c c c c c c c c c c
professional stamps. Plans of location, plant, cut and facade of the proposal, respectively bounded.	P CANCEL
Address & Contact	A
18 de Julio Av 1360, Montevideo Department 🛛 🍳	16
9876541	9
BOOKING	BOOKING



🗋 Users	× +						- 0 ×
← → C (D localhost/Citizen_GX16UpdCourse	e6.NetEnvironment/wwuser.aspx				4 2	o 🚯 :
Citiz	en Service					by G	eneXuš ^
Recents	Users						
	Users	Q Name				+	- INSERT
	Identification	Name	Photo	Age			
	20576	Ann Bert		43	UPDATE	DELETE	<u>.</u>
	11182	Brian Goals		77	UPDATE	DELETE	- 1
	89680	Cecilia Fernández	2	43	UPDATE	DELETE	- 1
	15587	Cecilia Lemos		16	UPDATE	DELETE	- 1
	81180	Eleonor Johnson		28	UPDATE	DELETE	- 1
	49128	Jessica Loyarte		30	UPDATE	DELETE	- 1
	12345	John Peters		45	UPDATE	DELETE	
	11245	Luciano Silveira	- 9 +	42	UPDATE	DELETE	
	49128 12345 11245	Jessica Loyarte John Peters Luciano Silveira		30 45 42	UPDATE UPDATE UPDATE	DELETE DELETE DELETE	

Le recomendamos abrir el Backoffice para ver los usuarios registrados y sus IDs:

Al ingresar uno y confirmar (por ejemplo el de ID 12345), se le presenta una pantalla donde se muestra foto y nombre del usuario ingresado, y se le permite seleccionar día y hora, dentro de los horarios habilitados del trámite, para reservarle un turno para ser atendido. Al hacerlo, si todo fue bien, se le presentará otra pantalla que le muestra toda la información del trámite, y se le ofrece la opción de ver todas las reservas que tiene hasta el momento:

🖬 🖉 🕑 📂 🛛 🔻 🖬 1:37		
← Enabling - Installing advertising ele	← Reservation Confirmation	← User Reservations
John Peters 🕑	Your date for Enabling - Installing advertising elements was reserved successfully! User John Peters	John Peters
Available hours	9/27/18 2:30 PM	9/27/18 2:30 PM Enabling - Installing advertising elements
From: 02:00 PM to 05:00 PM	Price	
Select a date	\$ 300	
Select date 💌 Time 💌	Address 18 de Julio Av 1360, Montevideo Department	
	Phone 9876541	
	Requirements 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 facade of the proposal, respectively bounded.	
CONFIRM	MY RESERVATIONS	CLOSE



En esta última pantalla también se visualiza foto y nombre del usuario, con exactamente el mismo diseño que antes, aunque donde se ha ocultado la posibilidad de cambiar de usuario.

Por otro lado, si vamos a la opción Complaints del menú, seleccionando el tipo de reclamo, el usuario podrá ingresar uno de ese tipo al sistema:



Observe que en esta pantalla, UserComplaint, también se muestra la información del usuario de la misma exacta manera que en la del trámite.

Si quiere modificar la manera en la que se muestra la foto (en las tres pantallas en las que aparece) para que no salga más redondeada, tal como se ve aquí:



▤◙∕⊙▻			V 4	2:09				
← Driver's license renewal, for cars a								
	John	Peters	C					
Available hours								
From	: 03:00 PM t	o 05:00 PN	1					
Select a date								
Select date	v	Time	ŧ	Ŧ				
	CONFIRI	м						

¿Cómo lo implementa?

SOLUCIÓN:

Abra el stencil UserProfile y vea que la foto tiene asociada la clase ImageUserProfile del theme CarmineSDCourse. Cámbiele la clase Border Radius, dejándola vacía.

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

Ejecute CitizenMenu y observe el cambio en las tres pantallas.



CREAR UN STENCIL Y USARLO

Ahora deseamos que la pantalla que muestra las actividades culturales (Activities) muestre arriba las actividades destacadas, y abajo todas las actividades. Cuando se selecciona una actividad en cualquiera de ambos grids, se visualiza una pantalla con su detalle:



Observe que la imagen de la actividad, su nombre y categoría se muestran de manera idéntica en el grid que muestra All Activities en el primer panel y en el segundo panel. Para evitar repetir el diseño es que utilizamos stencils.

Implemente un stencil para mostrar esta información y utilícelo en ambos paneles: Activities y ActivityDetail.

POSIBLE SOLUCIÓN:



Cree el stencil de nombre CulturalActivityHeader (puede hacerlo a mano, o puede posicionarse sobre la tabla que ya contiene la foto y los dos atributos en el panel Activities, y con botón derecho elegir Wrap as new stencil):

💈 CulturalActivityHeader * 🗙 🛅 CarmineSDCourse * 🗙	\sim	Properties	7 ×											
Layout * Variables Documentation		General Class												
MainTable Mi CulturalActivityPhoto		📜 🦺 🌾 🛛 Filter	×											
		v 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 \vee											
		Background	(none)											
		Visible	True											
		Invisible Mode	Keep Space											
		Enabled	True											
SD -	~													

Si lo hizo a mano: a la variable con la imagen configúrele las propiedades:

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

A la variable que muestra el nombre a la izquierda configúrele las propiedades:

- Label Position: None
- Class: AttributeActivityNameList

A la variable que muestra la categoría a la derecha configúrele las propiedades:

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

En el panel ActivityDetail, en la tabla donde insertaría los atributos CulturalActivityPhoto, CulturalActivityName y CulturalActivityCategory, que está vacía, busque el Stencil que acaba de crear, e insértelo.



S UserProfi	le 🗙 🚮 CarmineSDCare	ourse	× 📑 Activities * × 🛼 C iables Documentation	ulturalActivityH	leader X	Activity	yDetail 🗙		<>▽
 Application 	on Bar								
▲ Mair	Table Table2								+
Cultural	o o ActivityDescription	 	Insert Attribute/Variable Insert Button Delete SUPR Convert to Canvas Convert to Flex Wrap as New Stencil Swap table with Stencil Go To Event						
How do	I get there?	**	Insert Row Before						
now do		1	Insert Column Before						
Any Platfor	Image: mage: ma	i.	Insert Column After						Ť
Cutput		×	Delete Row						ŦΧ
Show : Gene	ral 📕 🗙		Delete Column				(-> (□ (+ A	utoscroll
OserProfile Layout Rules Applicat	CarmineSDCourse Events Conditions Vari elect Objects Filters Pattern:	X ables	Activities * X S CulturalActivit Documentation	yHeader X	ActivityDeta			General	Class
Cultura	Category: *ALL		•	Module/Folder	r:			-	•
How de	RCulturalActiv RCulturalActiv	ityPhoto ityName	8CulturalActivityCategory			&UserN	ame	•	
F) Output	Root Module CulturalActivit	yHead	ler	Root Modu	le UserProfile	Drofilo			
Show : Gene	30/3/2020 18:50 Cultural	Activi	y neader	8/8/20191	1:41 User	Profile			
	<							0	>
	1 selected object(s).						ОК	2 matching obj	ect(s). el



Nota:

• Deberá reemplazar las variables por los atributos CulturalActivityPhoto, CulturalActivityName y CulturalActivityCategory.

Si creó el stencil a mano, entonces en el panel Activities, posiciónese sobre la tabla del grid que muestra todas las actividades, y presionando botón derecho, observe que al hacer "Swap table with stencil":



y elegirlo, ya sustituirá la tabla por el stencil, respetando los atributos (no los cambia por variables).

Pruebe en ejecución (Run sobre el objeto CitizenMenu).

SMART GRID

El panel de actividades culturales muestra arriba un grid con las actividades marcadas como destacadas (ver atributo CulturalActivityFeatured de la transacción CulturalActivity). Se desea que esas actividades se vean en un formato carrusel, como se muestra en la imagen:





¿Cómo lo implementa?

SOLUCIÓN:	

	ŀ	Properties		.
Featured Activities	1	General Class		
GRID	[📜 🦺 🌾 🛛 Filter		\times
		 Grid: Grid2 		-
		Control Name	Grid2	
		Collection		
		Default Action	'ViewDetail'	
	۱.	Selection Type	Platform Default	
CulturalActivityCategory	L	Enable Multiple Selection	False	
	١.	Pull To Refresh	False	
All Activities	11	Inverse Loading	False	
GRID 👻	1	Default Selected Item La	(none)	
· · · · · · · · · · · · · · · · · · ·	- L.	∨ Control Info		
•		Control Type	SD Smart Grid	\sim
		Auto Grow	False	
		Scroll Direction	Horizontal	
CulturalActivityName CulturalActivityCategory		Snap To Grid	False	
		Items Layout Mode	Single	
		· · · · · · · · · · · · · · · · · · ·		

Cámbiele, también, el Width al Canvas para que ocupe el 50% y no el 100:



Activities * ×	\sim	• Properties	7 ×
Layout * Rules Events Conditions Variables Documentation		General Class	
- Application Bar	-	📜 🦺 🌾 Filter	×
		 Canvas: Grid2Canvas 	
Main Table Table1 E Grid2 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	~		

Observe en ejecución la diferencia con el horizontal grid.

FLEX GRID

Si observa el panel main del frontend web (de nombre CitizenWeb), además de ver las actividades culturales destacadas se ofrece una opción "See all Activities" para ver todas las actividades:





En esa otra pantalla queremos que el grid permita mostrarlas en forma variable, de acuerdo a su tamaño:



¿Cómo lo implementa?



SOLUCIÓN:

SeeAllActivities * ×	\sim	Properties	7 ×
Web Form * Rules Events Conditions Variables Help Documentation		General Class	>
Actions Insert MainTable Grid1	▶	Cell Spacing Appearance	2
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

Observe las diferencias con el Horizontal Grid que tenía antes.

BASE STYLE

Si edita los files de la KB encontrará que existe:

📰 Files 🗙	\sim
🕴 📰 Add New File 🔀 Delete Filter	
Files	-
ApercuPro-Regular	
ApercuPro-Bold	
SemanticUl_gxlibrary	

También encontrará una nueva propiedad "Base Style" en los objetos:

- Theme
- User Control



Esta propiedad permite asignar un estilo base a esos objetos.

En la propiedad se tomarán los valores de los Files de la KB que tengan extensión "gxlibrary" (son un zip que contiene todos los archivos css, js, assets, etc. provistos por los diseñadores o el Design System utilizado)

A nivel de theme encontrará la propiedad:

V Customization			
> 🔁 Localization	Properties	×	
✓ ➡ Themes	📜 🛃 🌾 Filter	X	
Carmine	Theme: CarmineWeb		
CarmineRTL		a	
> 🔽 🛅 CarmineSD	Name	CarmineWeb	
🔲 📑 Flat	Description	Carmine Web	
GeneXusXEv2	Base Style	Bootstrap v3	
C SimpleAndroid	Туре	Name	
C SimpleIOS	Туре		
🛐 CarmineSDCourse	Dark Theme	Bootstrap v3	
🔁 CarmineWeb	Color Palette	Bootstrap v3 RTL	
> Colors	Oualified Name	SemanticIII	
🚵 Images			
Files			

Y a nivel de objeto la veremos en seguida cuando creemos un User control.

USER CONTROL

Si ahora desde la página principal del frontend web elegimos un trámite:



Citizen Web x +		- 0 ×
← → C ③ localhost/Citizen_GX16UpdCourse7.NetEnvironment/citizenweb.aspx?		☆ ╗ ♀ ∰ :
Featured Activities Tango in the atrium every Saturday	Centenario Stadium	• Urban Bike Rental in Montevideo
		<image/>
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 renewal, for cars and trucks		
Driver's license from 75 years old		

De manera similar a la app mobile nativa, nos pide identificador de usuario para poder reservar hora para ese trámite:

🗅 Formality Reservation Web 🗙 🕂		- 🗆 X
← → C () localhost/Citizen_GX16UpdCourse7.NetEn	vironment/formalityreservationweb.aspx?DriverLicenseRenewal	☆ ╗ 0 🧶 :
Citizen Service		by GeneXus
Recents See All Activities — Citizen Web — Fo	ormality Reservat	
Ма	ke a Reservation	
User Identi	fication	
Forma Descri	lity Driver's license renewal, for cars and trucks pption	
Forma Requi	lity 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.	
Forma Price	lity \$ 400	
Forma Addre	ss 18 de Julio Av 1360, Montevideo Department	
From:	To:	
03:00 Forma Date T	PM 05:00 PM lity // 12:00 AM 25 lime	
Con	firm	



Al ingresar un ID de usuario válido y salir del campo, queremos ver la info del usuario de la siguiente manera:



Observe que esa Card se carga en el componente de nombre WCUserInformation que se carga en el web form:



FormalityReservationWeb ×
<no action="" group="" selected=""></no>
MainTable
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

¿Cómo implementa ese componente (WCUserInformation) que en su KB está vacío?

SOLUCIÓN:

Se trata de un control Card de SemanticUI. Si va al sitio de SemanticUI, y busca por Card, encontrará distintos tipos de estos controles. Elija por ejemplo la Card más simple y visualice el html, para poder copiarlo:





Aquí se lo damos escrito para que pueda copiarlo luego y no perder tiempo:

```
<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>
```

Cree un objeto User Control en GeneXus y copie en su "Screen Template" el html anterior:





Parametrice las propiedades, de acuerdo a lo que desea ver. En nuestro caso nos quedará así:





Aquí le damos escrito el código anterior para que pueda copiarlo:

```
<div class="ui card card-extra-width">
  <div class="image">
    <img src="{{UserPhoto}}">
    </div>
  <div class="content">
    <a class="header">{{UserName}}</a>
    <div class="meta">
        <span class="date">{{UserName}}</span>
        </div>
        <div class="date">{{UserBirthDay}}</span>
        </div>
        <div class="description">{{UserAddress}}</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>
```

Revise que la propiedad Base Style del objeto User Control tenga el valor "SemanticUI" definido. De lo contrario, seleccione ese valor.

En el componente WCUserInformation deberá insertar el user control que acaba de crear. En la Toolbox encontrará:



Arrástrelo al Web form y seleccione el User Control que creó antes:



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

Luego, inicialice las propiedades del User Control en base a los atributos:

WCUserInfor	mation * ×	\sim
Web Form Rul	es Events* Conditions Variables Help Documentation	
Events	~	
1 = Eve 2 3 4 5 6 7 8 - Ene	ent Load UserInformationCard1.UserPhoto = UserPhoto.ImageURI UserInformationCard1.UserName = UserName UserInformationCard1.UserBirthday = "Born: " + UserBirthDate.ToString() UserInformationCard1.UserAddress = "Address: " + UserAddress UserInformationCard1.UserEMail = "E-Mail: " + UserEmail	- ¹ 11
10		>

Nota: Para corroborar que se haya cargado correctamente la Base Library (en este caso SemanticUI), revisar el directorio web de la KB (Tools → Explore Target Environment Directory) y ver si existe el folder "SemanticUI". Si no existe, sugerimos realizar un Rebuild del objeto CitizenWeb.

Ejecute.



AGREGAR UNA ACCIÓN AL USER CONTROL

Opcionalmente, puede querer que se permita desde la Card editar la información del usuario.



Eso no venía exactamente así en el html source del control de SemanticUI. Puede personalizar su objeto User Control, agregándole:





Y luego en el Web component:

🔐 UserInformationCard X 👼 WCUserInformation* X	\sim
Web Form Rules Events* Conditions Variables Help Documentation	
Load ~	
<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"</pre>	- ¹ 1,1 ini
7 - Endevent 8	>

Pruébelo en ejecución.

Falta programar el evento. Para ello:



UserInformationCard × 🛃 WCUserInformation ×	\sim
Web Form Rules Events Conditions Variables Help Documentation	
UserInformationCard1.OnClick ~	
<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>	
<pre>8 9 Event UserInformationCard1.OnClick 10 User(TrnMode.Update, UserIdentification) 11 Endevent </pre>	~

Vuelva a probar.

KB SOLUCIÓN

Podrá descargar de GeneXus Server la KB solución de este práctico para comparar resultados. Es la versión de la KB de nombre 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

(598) 2601 2082 (52) 55 5255 4733 (1) 201 603 2022 (55) 11 4858 0300 (81) 3 6303 9381 (81) 3 6303 9980