Modelo conceptual de Smart Devices





Las aplicaciones para Smart Devices organizan y manipulan la información estructurada de las **entidades** de la realidad con las que se trabaja (por ejemplo, la realidad podría ser la de una agencia inmobiliaria).

Subyace una estructura de interacción que podemos conceptualizar de la siguiente forma:

- Por cada entidad (por ejemplo "propiedad inmobiliaria" representada por la transacción correspondiente) se puede:
 - Elegirla de un menú y visualizar la <u>lista</u> de elementos que la constituyen en ese momento ("lista de propiedades inmobiliarias" en nuestro caso)
 - o ver la información detallada de un elemento en particular de la lista (en nuestro caso la información de una propiedad) y realizar alguna acción sobre el elemento (como por ejemplo actualizarlo).

¿No era principalmente esto lo que implementábamos con el patrón Work With de los modelos GeneXus Web?



Podemos ver las **entidades** de la realidad (por ejemplo, la dela agencia inmobiliaria) como **recursos** que necesitamos manipular.

Cobrarán importancia en este modelo las **transacciones** como business components, los **data providers** que devuelven datos estructurados y los **procedimientos**.

Desarrollar una aplicación para Smart Devices implicará principalmente trabajar con el **patrón** work with específico para este generador, que a partir de las entidades (transaccciones) elegidas construirá todo lo necesario para manipular la información: esto es, construirá el **listado**, el **detalle** de un elemento, y las acciones ante un evento disparado por el usuario...

en los lenguajes específicos de cada plataforma.



Por otra parte necesitaremos de un objeto del estilo de los generados por el work with, pero más flexible, no necesariamente asociado a una transacción, que nos permita por ejemplo pedir datos al usuario, como una fecha y realizar alguna acción. Será un objeto similar a un web panel.

Y por último, necesitaremos de los dashboards, objetos que cumplen la función de menús, utilizados principalmente como punto de entrada de la aplicación.

Resumiendo: implementar una aplicación para Smart Devices con GeneXus implicará trabajar con el patrón Work With para navegar por la información relacionada y manipularla, crear paneles para hacer consultas, desplegar información o pedir datos al usuario tal como se hace con los paneles web, y crear menús.



Por último, tal como ocurría en las aplicaciones web, el modelo conceptual automáticamente vincula las navegaciones de entidades que se encuentran estructuralmente interrelacionadas.

Por ejemplo, teniendo definidas las transacciones Neighborhood y Property, para representar respectivamente los vecindarios con los que trabaja la inmobiliaria y las propiedades inmobiliarias que maneja, cuando ambas entidades se relacionan, automáticamente el pattern work with lo refleja...

Veámoslo...

En nuestra KB tenemos las transacciones Property y Neighborhood, por el momento no relacionadas. Tenemos aplicado el pattern Work With en ambas.

Y tenemos un objeto Dashboard, que invoca por un lado al Work With de propiedades inmobiliarias (vemos que está invocando al Level Property y específicamente dentro del Work With Level Property, al list... y por otro lado vemos que está invocando también al work with de vecindarios. También a su nodo List.)



Observemos detenidamente este work with para ver cómo cambiará cuando interrelacionemos luego las entidades.

Podemos ver que el nodo List corresponde a esta pantalla en la cuál dentro de un grid tenemos los atributos del neighborhood del vecindario que queremos mostrar.

🐼 Start Page 🗙 📑 Re	alEstateApp × Property × Reighborhood ×		
Patterns usable in this object (u	underlined means pattern is applied)		
🝠 Category 🛛 😼 Work With	Mork With for Smart Devices		
Apply this pattern on save			
Level (Neighborhood)	Action Bar Insert		
Detail	e Mage and the second s		
Any Platform - Any Size - View - Any Orientation - Delete Layout			
I Structure I Structure			

Así en ejecución tenemos: (aquí estamos viendo el dashboard):...



Tenemos aquí el listado de los vecindarios.

👪 📶 🕶 5:54 PM
Work With Neighborhood
Coral Gables
Fort Lauderdale
North Miami

Luego eligiendo uno, vemos la información detallada del mismo, cuyo correlato se encuentra en la sección general.

	- #1 🗐 🕑	5:55 PM
Coral Gables		
Id 3		
Name Coral Gables		

Las propiedades son independientes. Si observamos el detalle de una de ellas vemos su información general sin ninguna vinculación con el vecindario. Ahora establezcamos la relación.

Vamos a la estructura de propiedades y agregamos a NeighborhoodId... pidiéndole que acepte nulos... y a NeighborHoodName. Grabamos.

🐼 Start Page 🗙 🗄 RealEstateApp	🗙 🔲 Property * 🗙 📃 I	Veighborhood 🗙		-
Name	Туре	Description	Formula	Nullable
🖃 📰 Property	Property	Property		
- 💡 PropertyId	Id	Property Id		No
🔑 PropertyName	Name	Property Name		No
··· • PropertyListingDate	Date	Property Listing Date		No
 PropertyAddress 	Address	Property Address		No
	Image	Property Front Image		No
PropertyOperation	Operation	Property Operation		No
NeighborhoodId	Id	Neighborhood Id		Yes 🖂
NeighborhoodName	Name	Neighborhood Name		
🖪 Structur 🖬 🏍 Web 🗐 😽 Win F.	🗟 Rules 🕅 Events 🚱 Va	ariables 🥏 Help 👫	Docume	Patterns

Y veamos qué sucedió con el WorkWith de neighborhood. Está apareciendo una nueva sección. Como vemos está mostrando un grid con las propiedades inmobiliarias. ¿Cuáles? Las de ese vecindario elegido.

🖉 Start Page 🗙 📑 Rea	alEstateApp 🗙 🖬 Property 🗙 🖬 Neighborhood 🗙 🗸 🔻	
Patterns usable in this object (u	nderlined means pattern is applied)	
🝠 Category 🛛 😨 Work With	Work With for Smart Devices	
Apply this pattern on save		
Level (Neighborhood)	Action Bar	
Detail	GRID	
Section (Property)	PropertyFrontIma PropertyName	
	Ze Z	
l	5	
	Any Platform - 📓 Any Size - 📓 View - 📓 Any Orientation - Delete Layout	
😺 Layout 🗟 Rules 😰 Events 🝸 Conditions 🚱 Variables		
🞚 Structure 🗟 🐔 Web 🗟 🦟 Win F 🗟 Rules 😰 Events 🗟 Variables 🦃 Help 🗟 Docume <table-cell-rows> 👪 Patterns</table-cell-rows>		

Veámoslo en ejecución. F5.

Reorganicemos.

🚳 RealEstateAgency GeneXus X							X
File Edit View Layout Insert Build Kn	owledge Manager Window	Fools Help					
🛯 🖉 🕹 🕰 🗠 🗠 🛬 📲	🗄 🛗 🚵 🕨 🛛 Release	*					
🖆 Knowledge Base Navigator 🛛 📮 🗙	Start Page 🗙 🗐		Neighborhood × Eimpact Analy	vsis X -	Properties		4 ×
Folder View	Database needs to be	e reorganized.			🔡 👌 Filter		
Quick Access	This report describes Database	changes and how they will be handled by reor	panization programs.		Table: Property		
RealEstateAgency	Please select Reorganize to pro	ceed or Cancel.			Name	Property	
Main Programs	03 December Canad				Description	Property	
🖶 🛅 Objects	- I neorganize Cancer				Cluster index		
🕀 🚞 GeneralWeb	📀 🛄 Property	Table Property specification		×	Storage area		
E SmartDevicesApi		Table name: Decent.			Text storage area		
Gx0010		Table name: Property			Initial size	0	
A Messages		Table Chevelon		E	First extent size	0	
- Reighborhood		Attribute	Definition	Previc	Next extents percentag	0	
- WorkWithDevicesNeighborhoo		PropertyId	Numeric (4)Not null Autonumber	FIEVIC	Minimum number of exte	0	
Property		PropertyName	Character (30)Not null		Change frequency	0. Pretty Often	
RealEstateApp		PropertyListingDate PropertyAddress	Date Not null				
P-B Tablee		PropertyFrontImage	Image Not null				
Customization		PropertyFrontImage GXI	Varchar (2048)Not null				
Cocumentation		PropertyOperation	Character (1)Not null				
- D Files		New Neighborhoodra	Numeric (4)				
		Indexes	Definition	Cor			
		TPROPERTY	primary key				
		•		•			
	🛿 🔂 0 Errors 🧘 0 Warnings	🚭 1 Success					
	Output			4 ×			
	Show: Build	- X		+ Autoscroll			
	Processing			*			
	Searching for new or changed tab	oles					
🔄 Folder View	Analyzing Property Removing specification files						
Category View	Specification files removed. Generating messages for English						
🗟 Latest Changes View	Generators messages generated.						
Preferences	Searching for deleted tables			=			
	Loading table and attribute proper Saving specifications	ties			Categories		
÷				-			
🛾 Knowledge Base 🔲 Document Outlin	Output Breakpoints	P Search			🔆 Toolbox 🛛 🖧 Indexer I	Monitor Properties	
C:\Models\RealEstateAgency_ Done				L.	n 1 Col 1 Ch	1	INS
🚱 Coogle 🚞 🚺	📀 🖪 🥳	ž 🕂 🐼 🔜				13 12 12 04:04 13 12 10 09/02	+ p.m. 2/2012

Luego, una vez asignados los vecindarios a las propiedades que teníamos ingresadas, vayamos a ver la lista de vecindarios y elijamos ver el detalle de uno de ellos.

🏭 📶 🖾 6:15 PM	🏭 📶 🛃 6:15 PM
Fort Lauderdale	Fort Lauderdale
Name Fort Lauderdale	Name Fort Lauderdale
General Property	General Property
Id 2	Utopía
Name Fort Lauderdale	

Podemos ver que aparecen dos tabs que coinciden con las secciones que vimos del pattern. El primer tab corresponde a la Section General que contiene la información general del vecindario y el segundo tab corresponde a la otra sección que muestra las propiedades inmobiliarias de *ese* vecindario. Veamos qué ocurre si elegimos una de ellas...

📑 📶 🕝 6:17 PM
Name
Utopía
7/21/11
Address 🥚
3975 Utopia Ct, Maimi 🕴 🕴 📍
Front Image
Operation Sale
Neighborhood Id 2
Neighborhood Name Fort Lauderdale

Nos está llevando al detalle del work with correspondiente a las propiedades, mostrándonos el detalle de esa propiedad y entre su información podemos ver el vecindario, que si seguimos este ícono, nos llevará, cíclicamente, nuevamente, a la información general del vecindario.

	🌇 📶 🕝 6:18 PM
Fort Lauderdale	
Name	
Fort Lauderdale	
General	Property
Id	
2	
Name	
Fort Lauderdale	



Como pudimos ver, si las entidades están relacionadas, las navegaciones ofrecidas también se interrelacionan, sin necesidad de hacer nada.



Con esto hemos visto una pequeña muestra de lo simple y poderoso que es el modelo conceptual.



Qué sigue?

Desea ver cómo personalizar el work with para darle más potencia y adecuarlo a sus necesidades?

Desea ver cómo crear paneles que le permitan extraer y presentar datos de forma flexible, pedir datos al usuario, etc.?

Desea descubrir la arquitectura subyacente?

Continuará...

