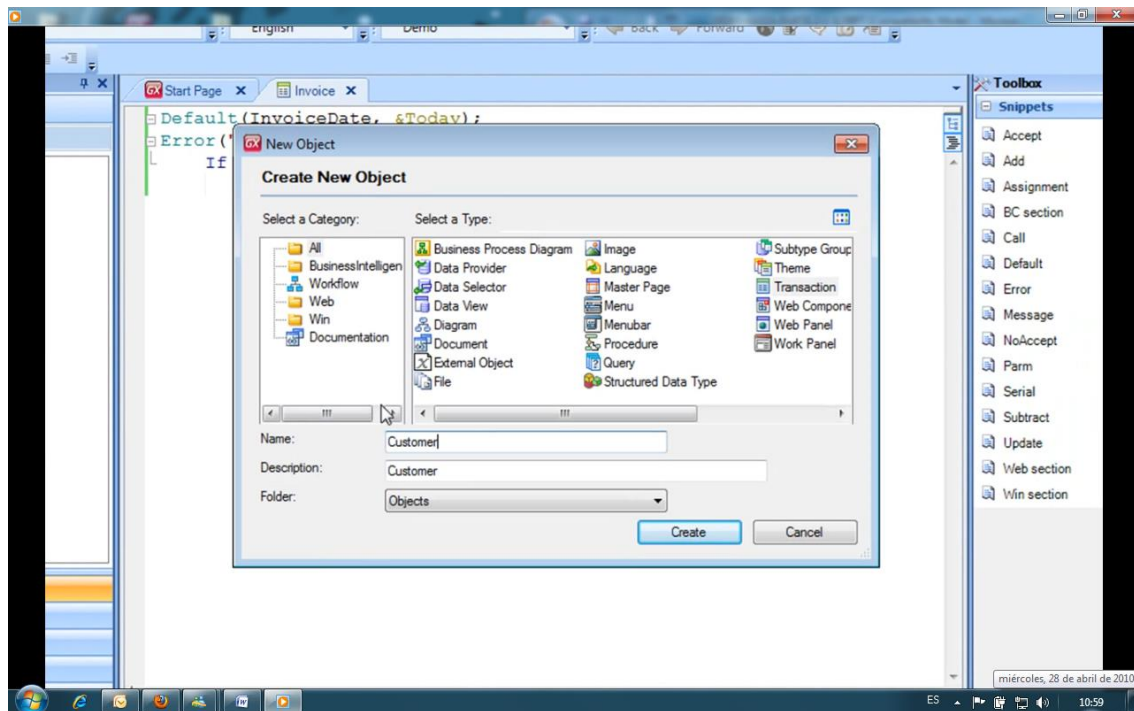
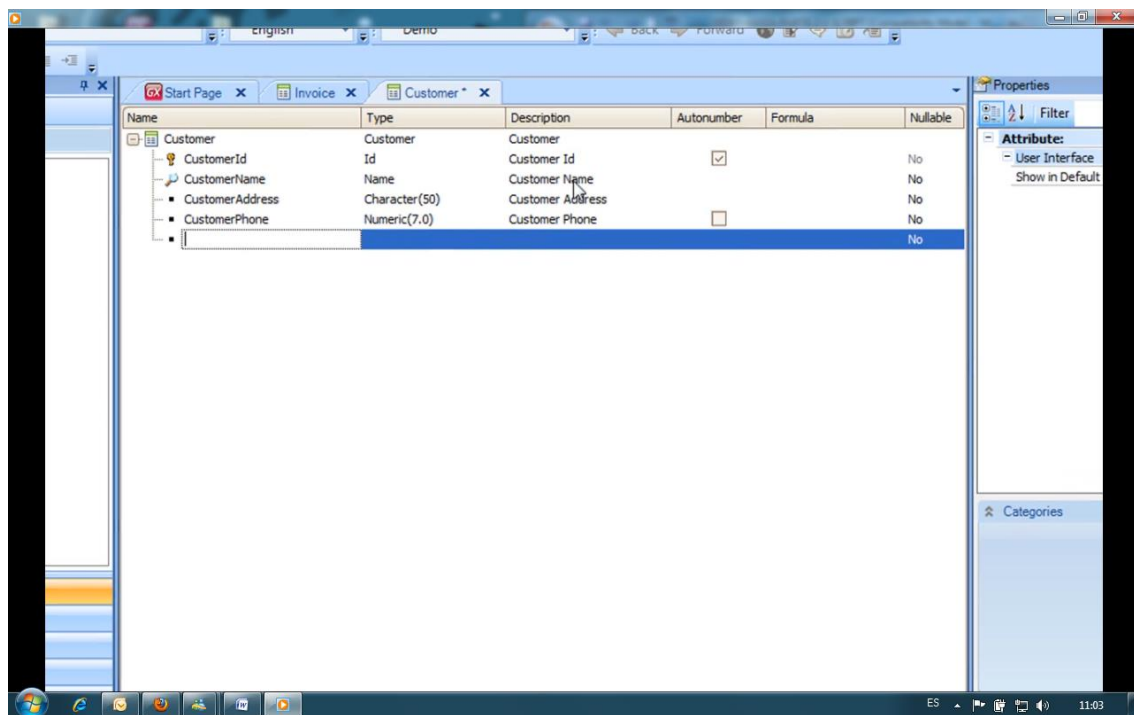


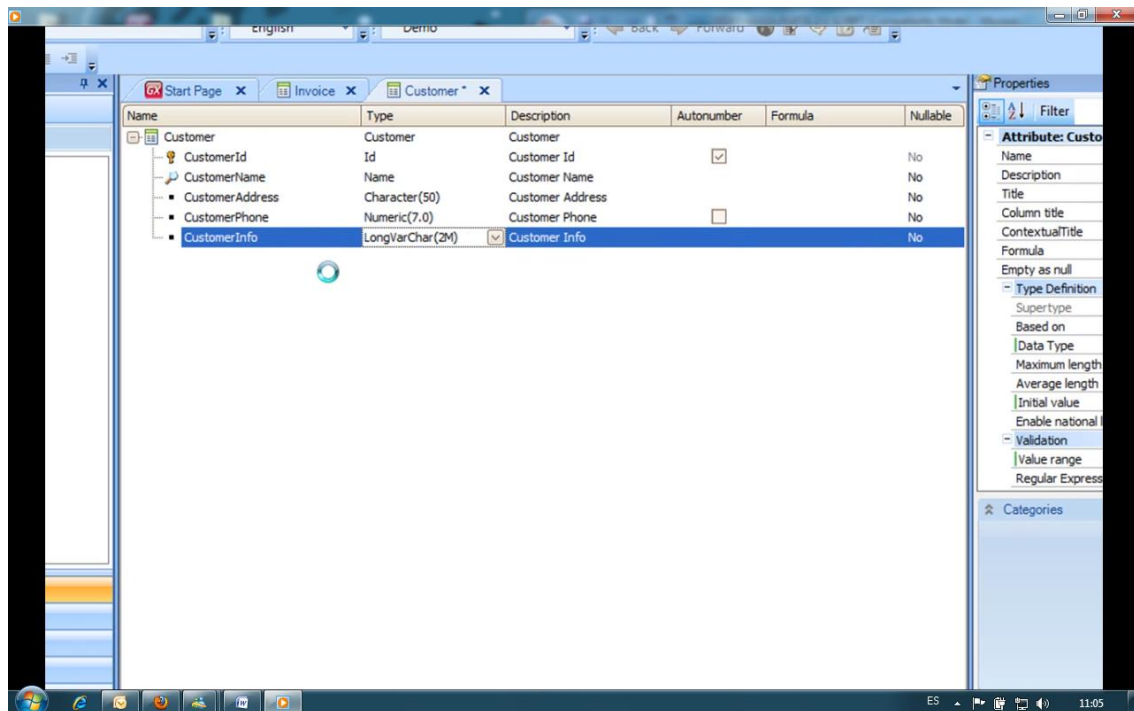
Overview GeneXus - Demo: 3. Se agregan transacciones para registrar Clientes y Productos al sistema de facturación.



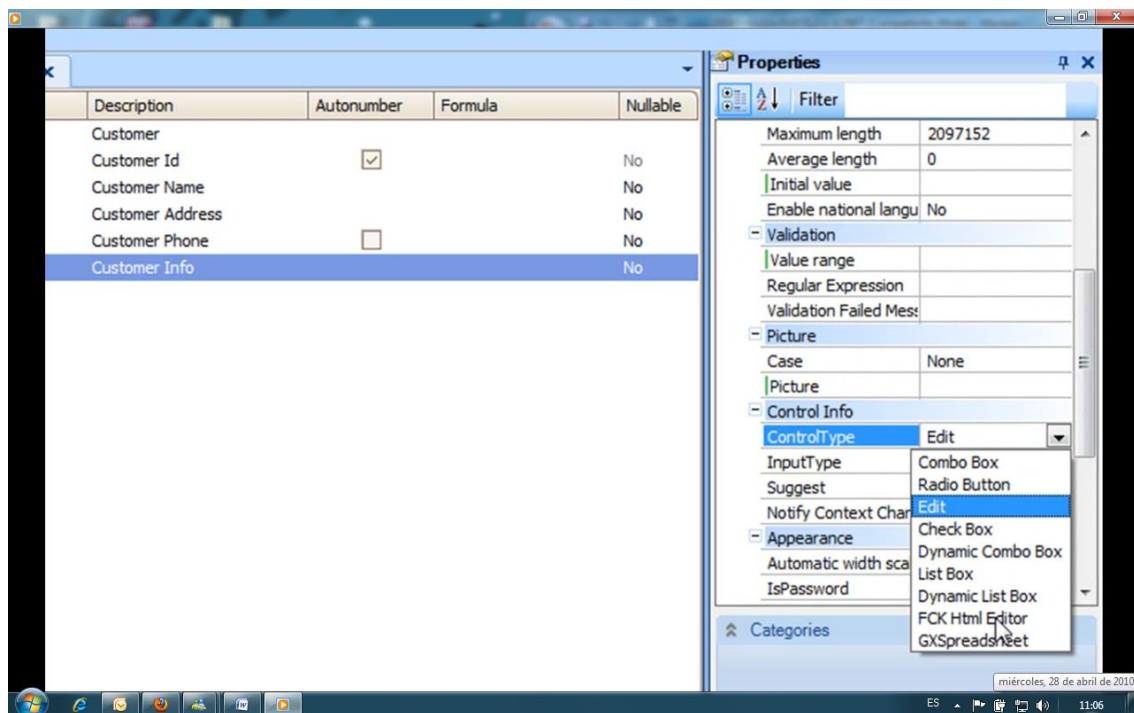
Vamos a comenzar creando una transacción cliente.



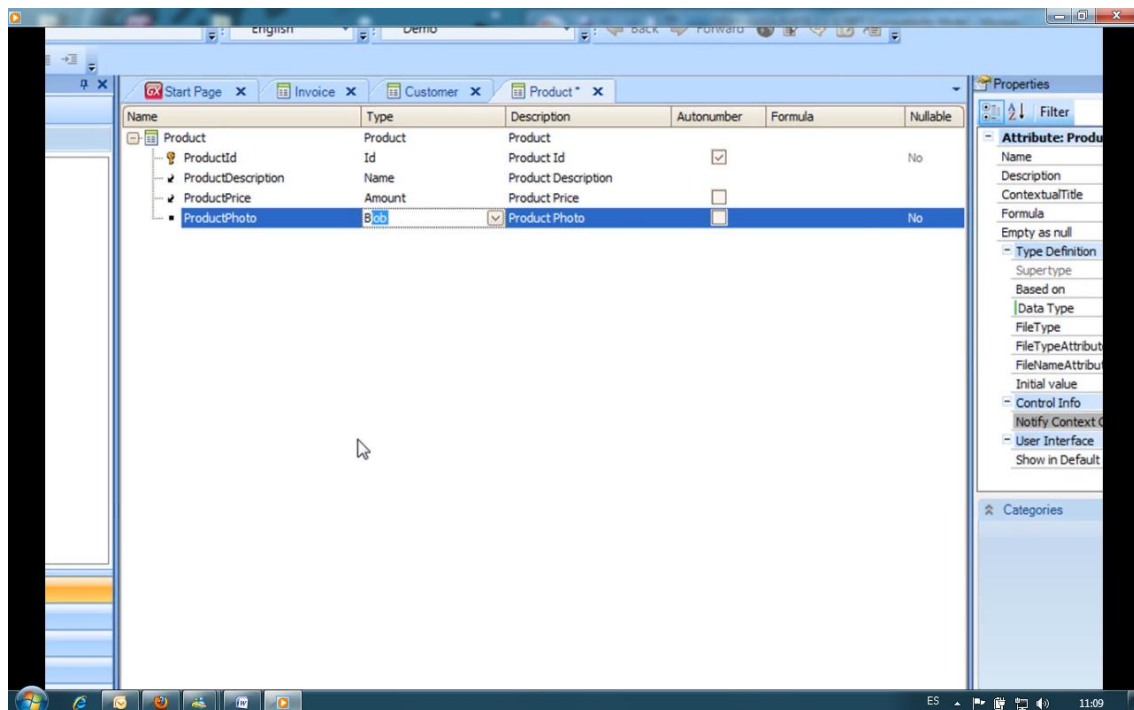
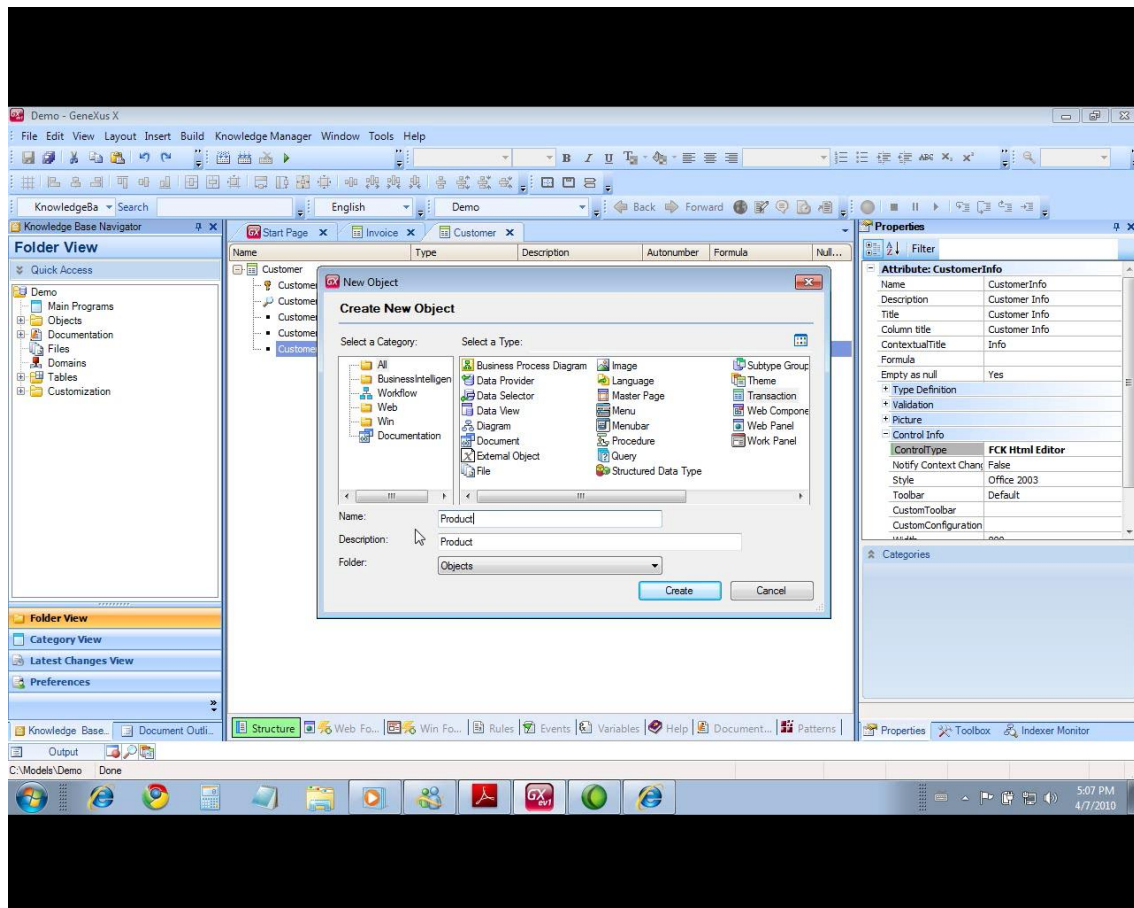
Y un cliente **tiene**: un identificador, que esta auto numerado; tiene un nombre, tiene una dirección, un teléfono



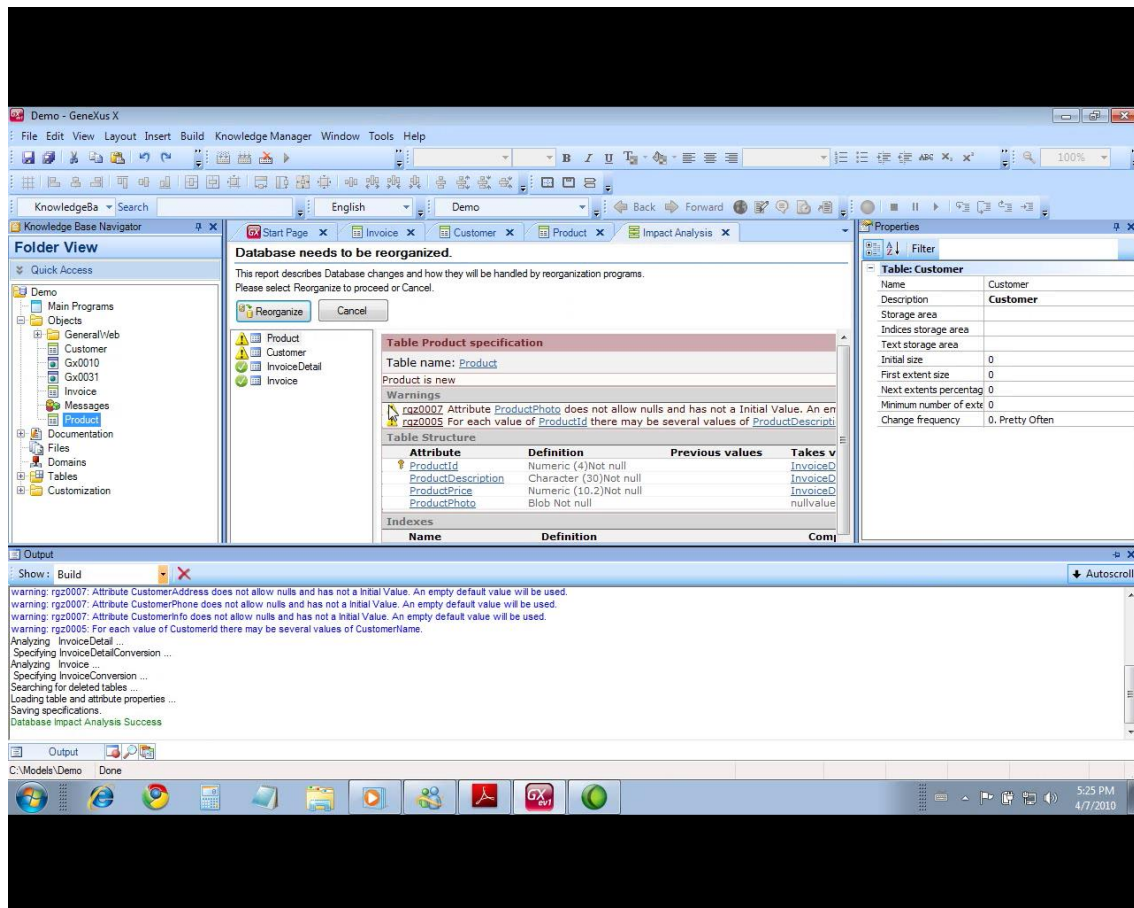
y un atributo info donde vamos a agregar cualquier comentario o información que queramos de ese cliente.



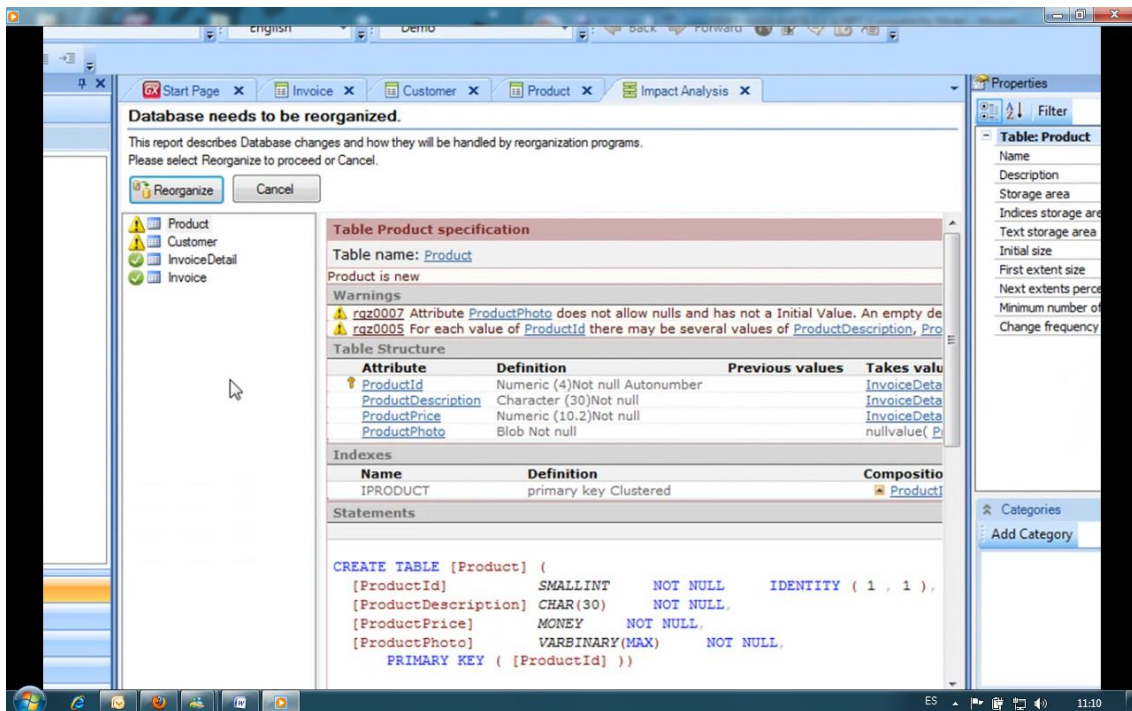
Y en particular a esta información vamos a agregarla en un editor de texto enriquecido. Salvamos y vamos a agregar ahora la transacción de producto.



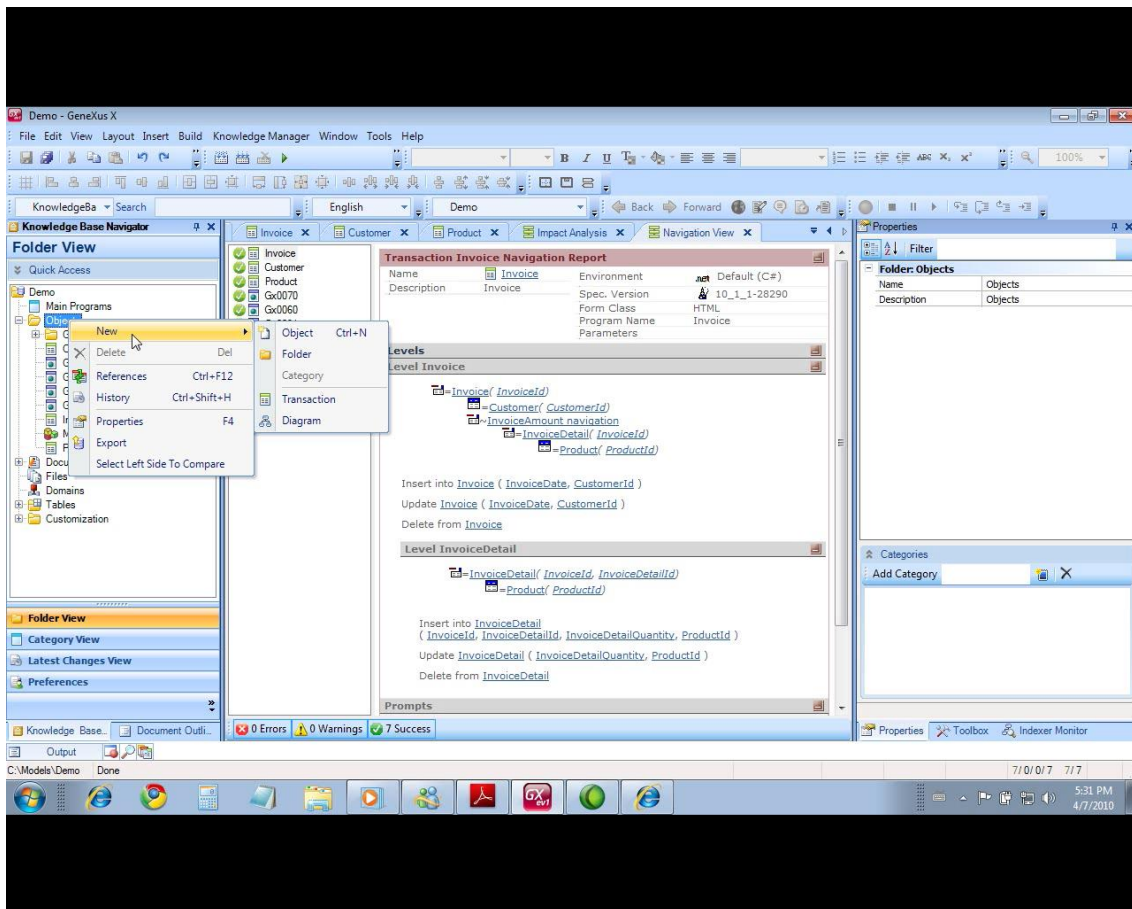
Un producto tiene un **Id**, una **descripción**, el **precio**, observen que a medida que voy ingresando los atributos ya me los va mostrando y los voy eligiendo de allí. Y un producto también tiene una foto. Vamos a ejecutar



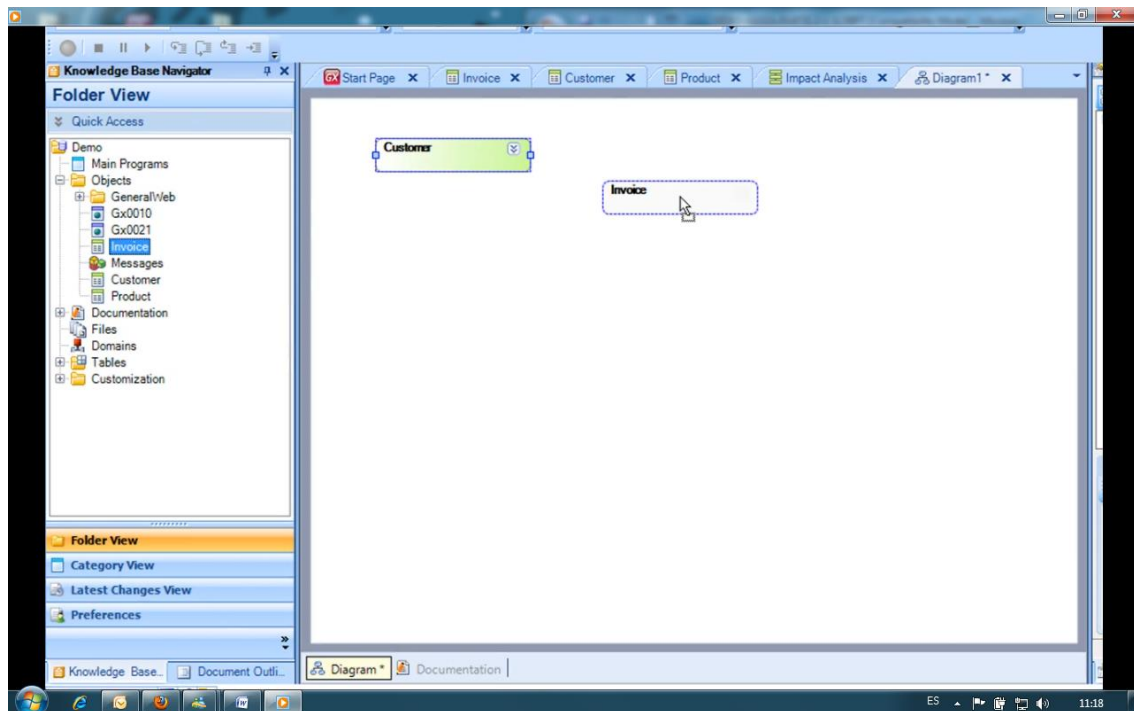
con lo cual **Genexus** hará un análisis de impacto de los cambios que deberá hacer a la base de datos y



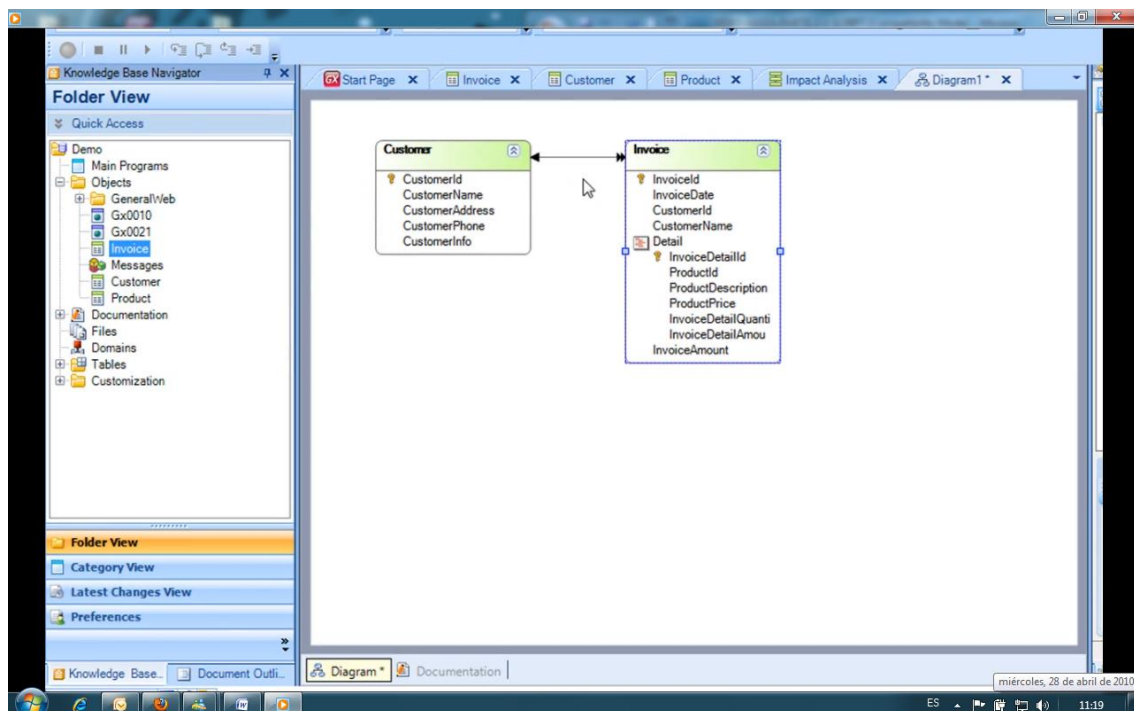
aquí me está proponiendo reorganizar y me va a crear entonces una tabla de productos y una tabla de clientes. Podemos ver **esto** si hacemos un diagrama de transacciones y tablas.



Vamos a crear un objeto diagrama.

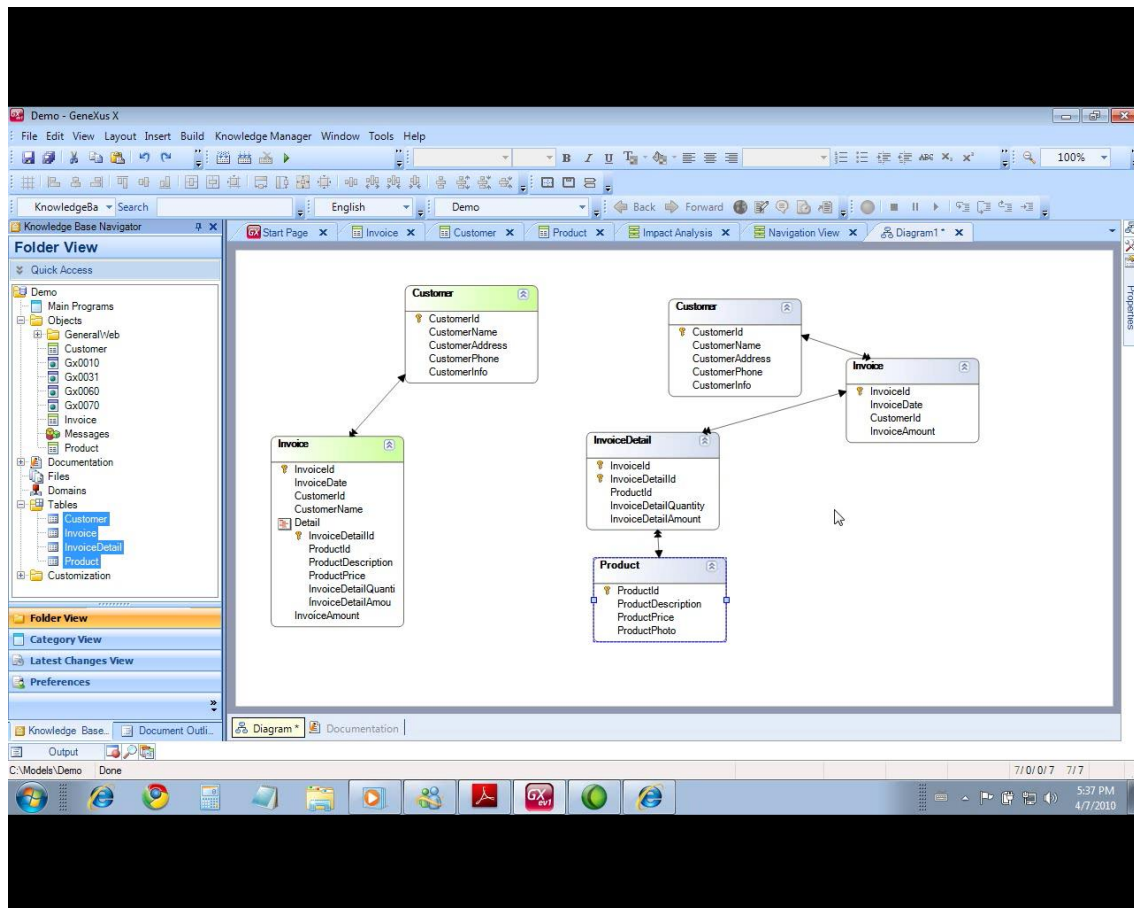


Vamos a arrastrar la transacción de Customer y de Invoice



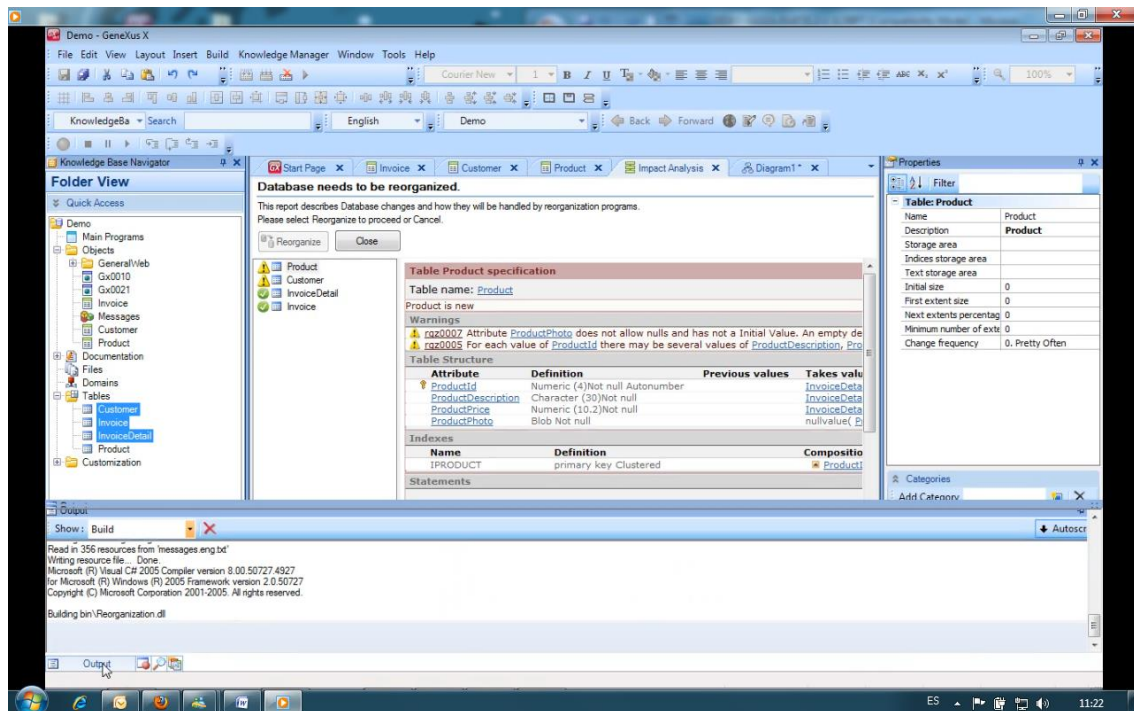
y observemos que Genexus establece automáticamente una relación de 1 a N entre ambas entidades.

Ya que el atributo CustomerId que aquí es clave primaria, aquí está como clave foránea. Vamos a ver ahora las tablas que se crearon a partir de estas 2 transacciones.



Vemos que el atributo **CustomerName** que **está** en la transacción de **Invoice** no **está** en la **tabla** de Invoice ya que este atributo fue eliminado por la **normalización**.

Cuando diseñamos transacciones no tomamos en cuenta las estructuras de las **tablas** sino más bien la información que el usuario necesita ingresar en las pantallas de su aplicación.

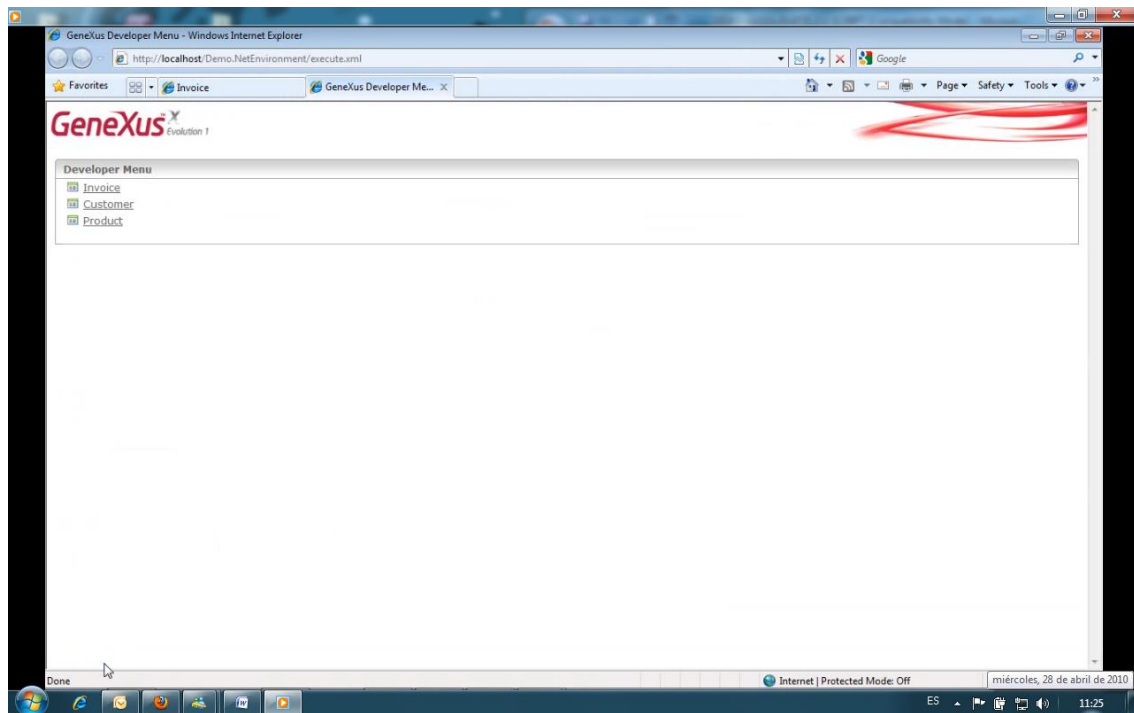


Vamos a reorganizar y mientras tanto un par de consideraciones importantes.

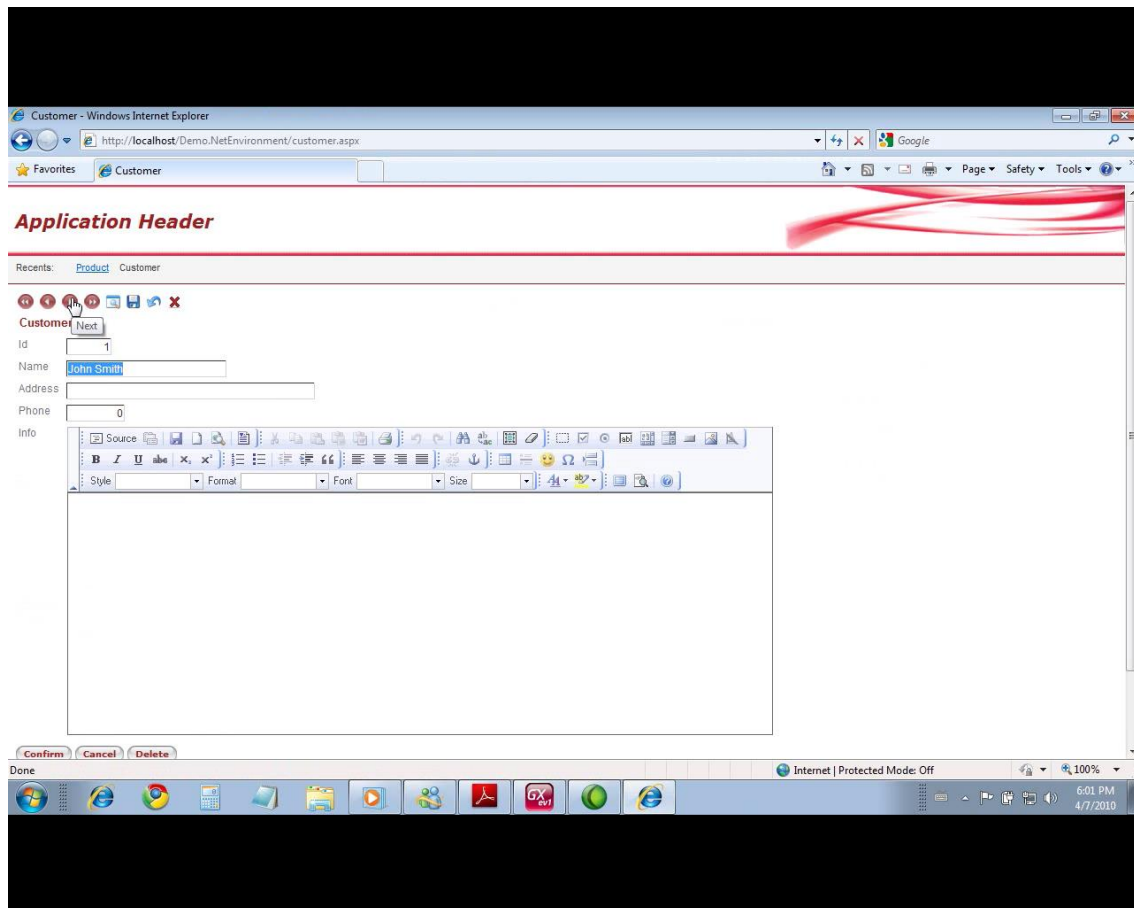
Como habrán **observado**, cuando ingresé las transacciones de clientes y productos puse los mismos nombres de atributos que use en la factura.

Genexus considera que los atributos que se llaman igual corresponden al mismo concepto y por lo tanto se va a dar cuenta que el CustomerId de la transacción facturas es el mismo de la transacción cliente.

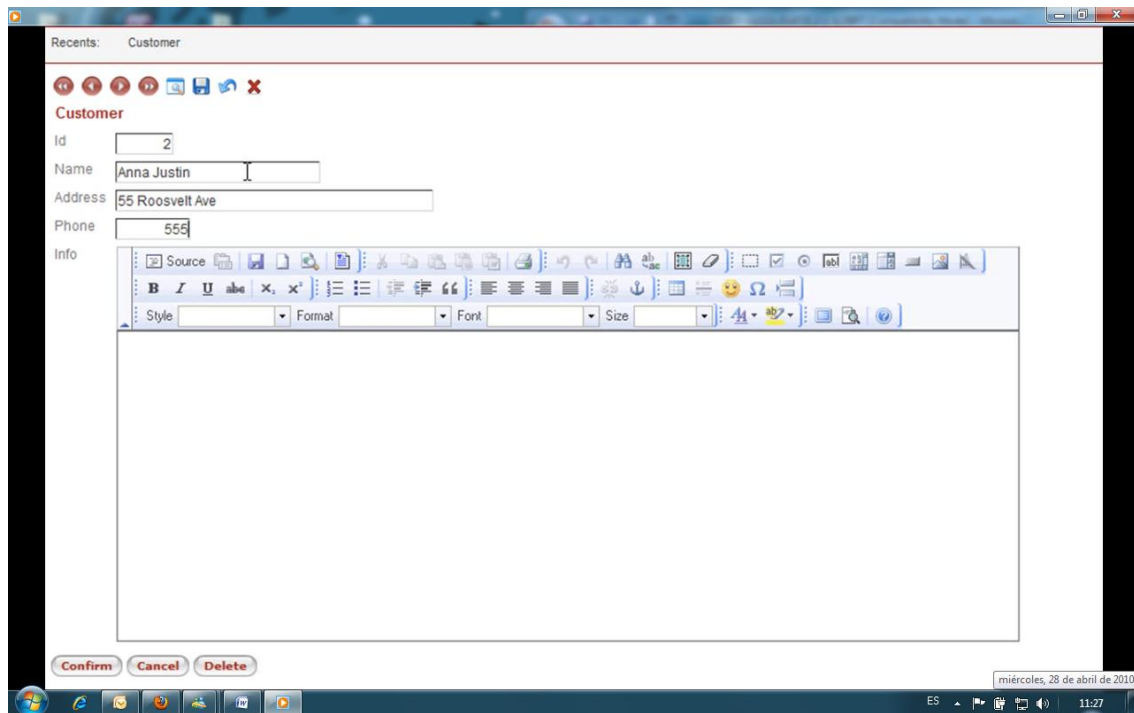
A partir de eso establece las relaciones del modelo de datos y normaliza las tablas en tercera forma normal.



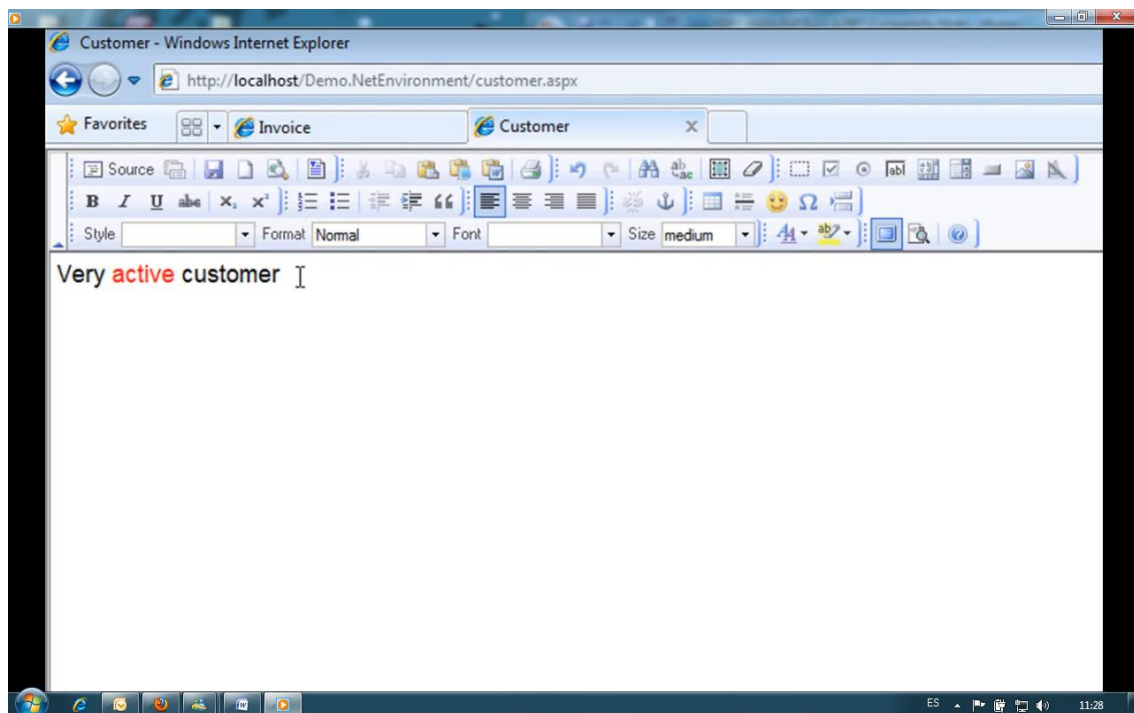
Vamos a agregar algunos datos de **clientes y productos**



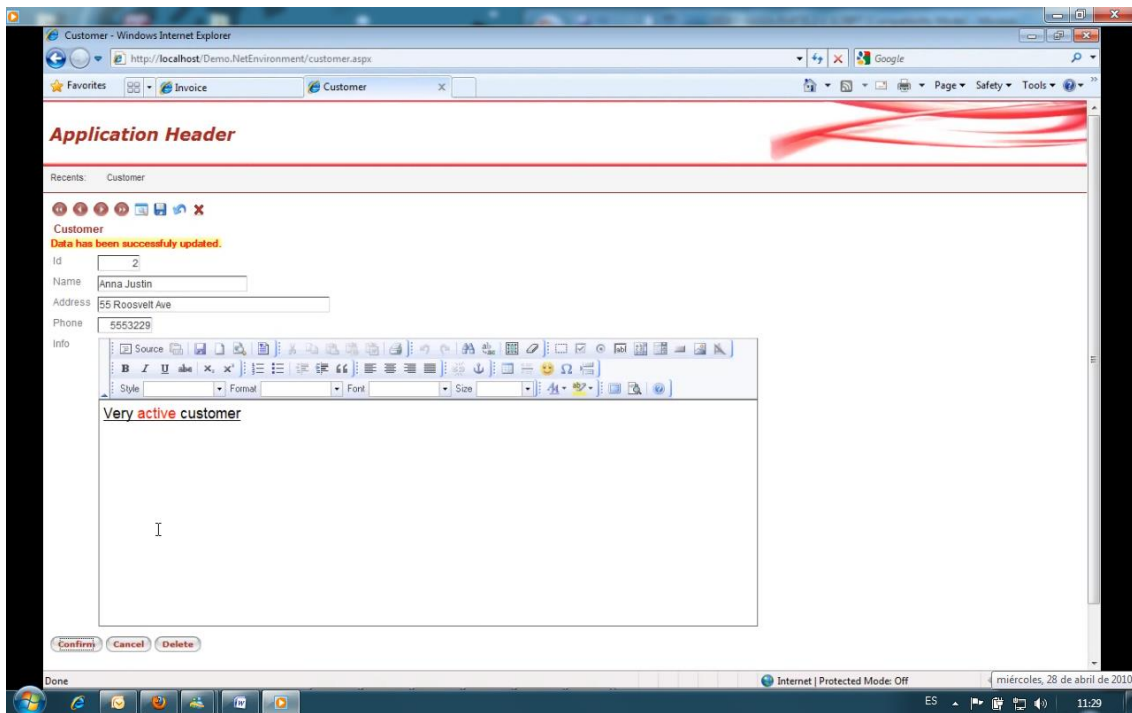
Observamos que los **clientes** ya están cargados, esto significa que ante un cambio de la realidad nosotros cambiamos la **aplicación** y Genexus **cambió** el modelo de datos pero sin pérdida de información y esto es muy importante porque **Genexus** nos asegura que si una aplicación **estaba** funcionando, luego de hacerle cambios estructurales **seguirá** funcionando.



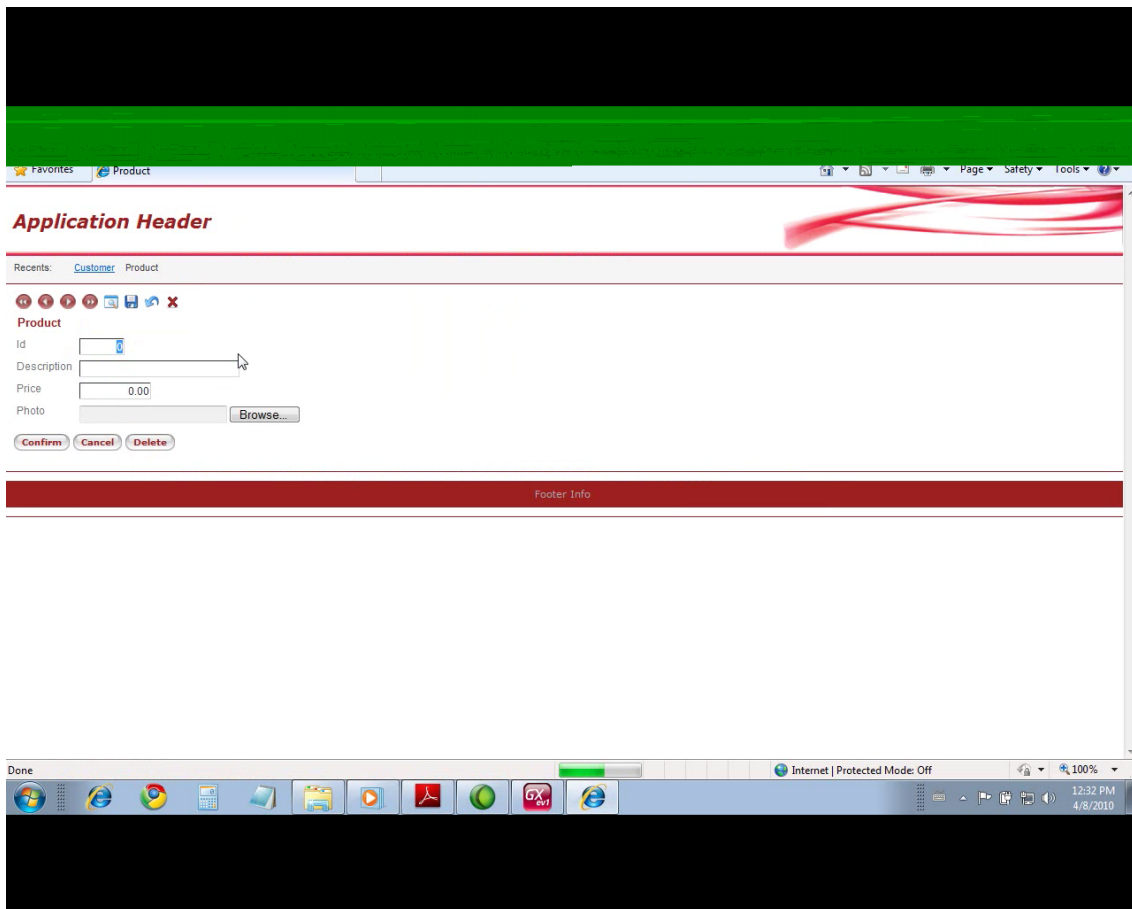
Vamos a completar los datos de este cliente y aquí en este editor de texto embebido en nuestra aplicación podemos

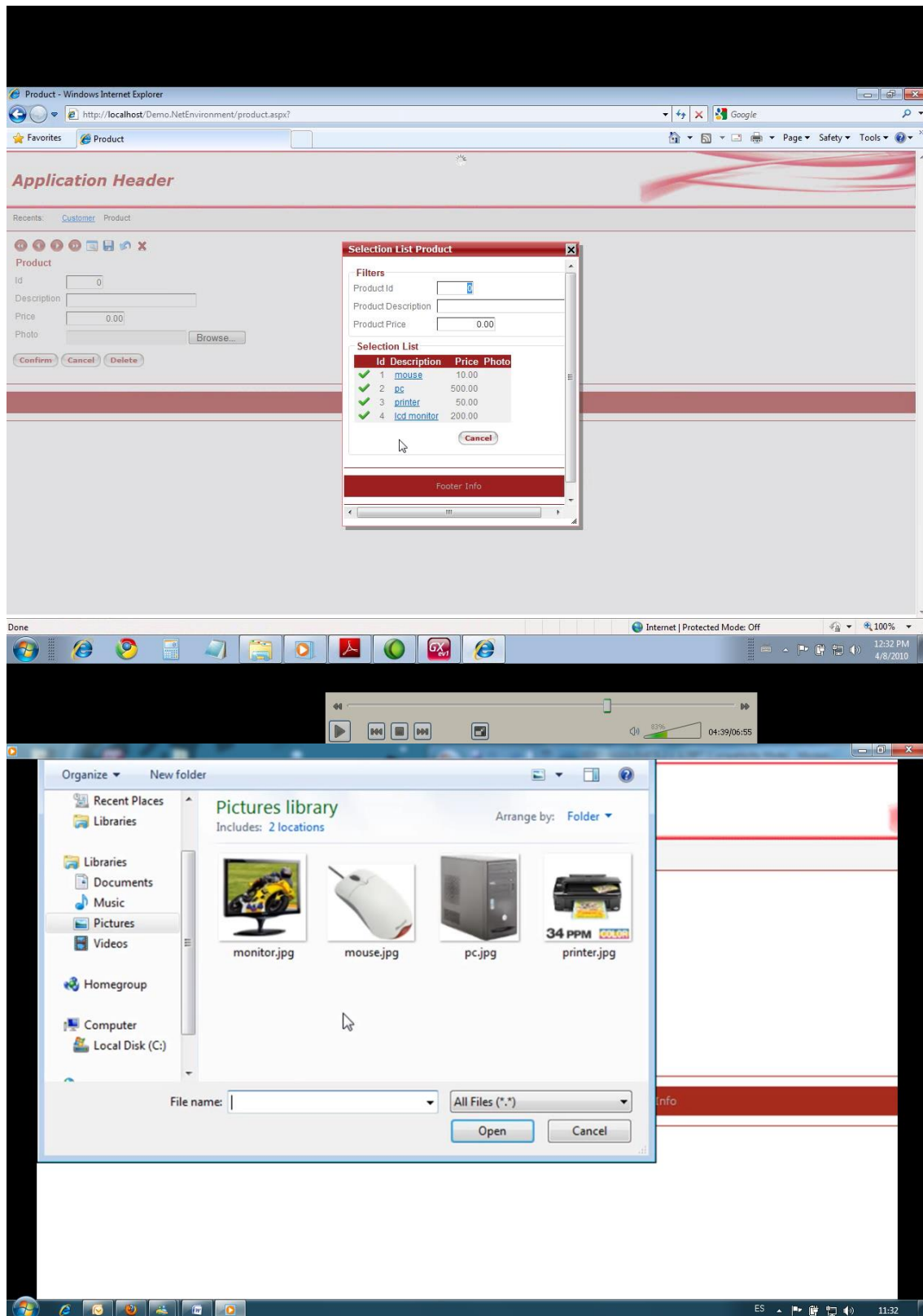


agregar cualquier comentario que nos parezca conveniente y realzar la **información** de la manera más apropiada utilizando este formato enriquecido.

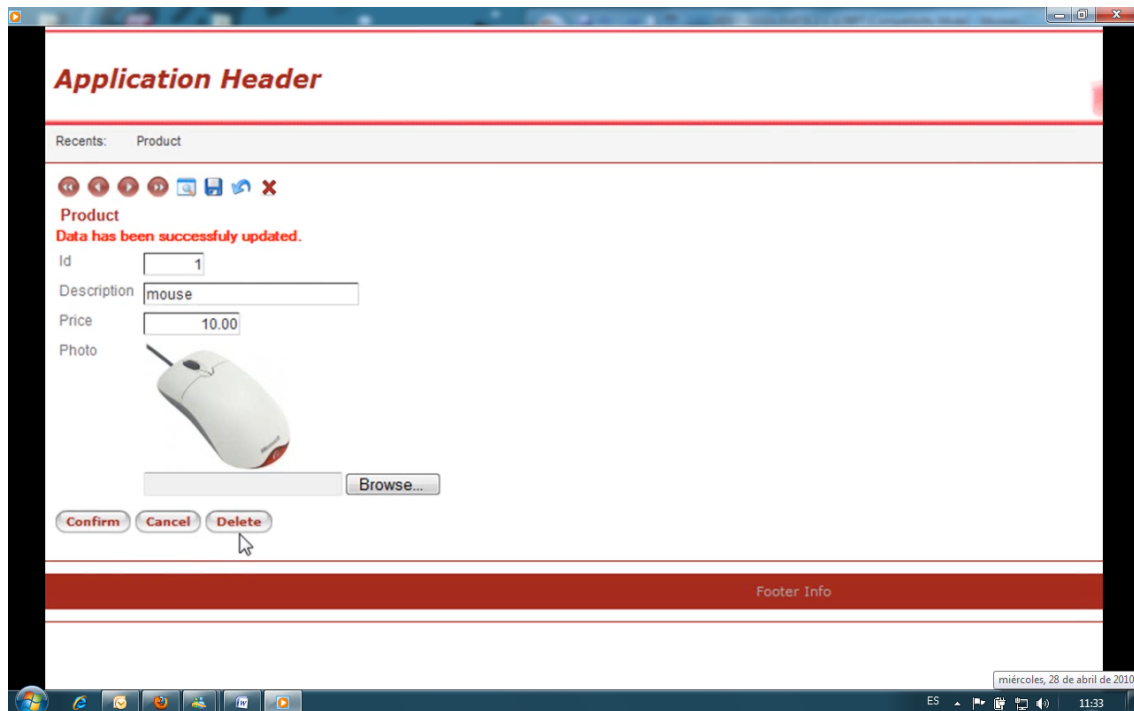


Vamos a confirmar a este cliente y completar algunos datos de productos.

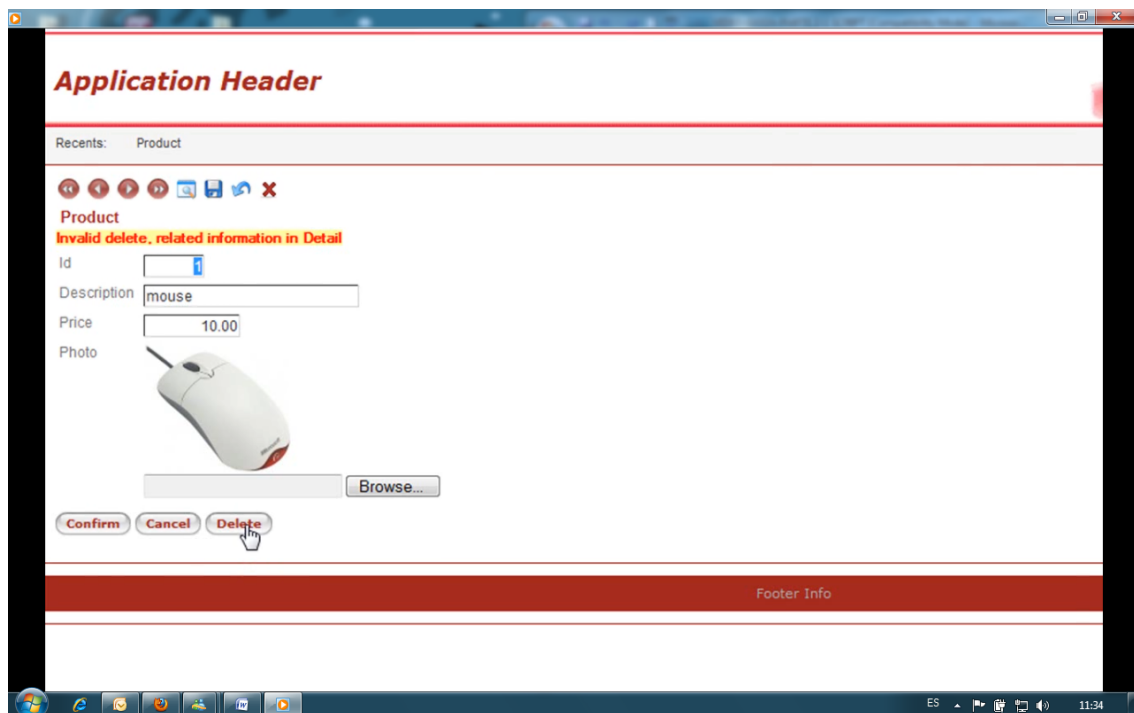




Como ya dijimos los productos ya **están** cargados, así que vamos a agregar una foto para el mouse.

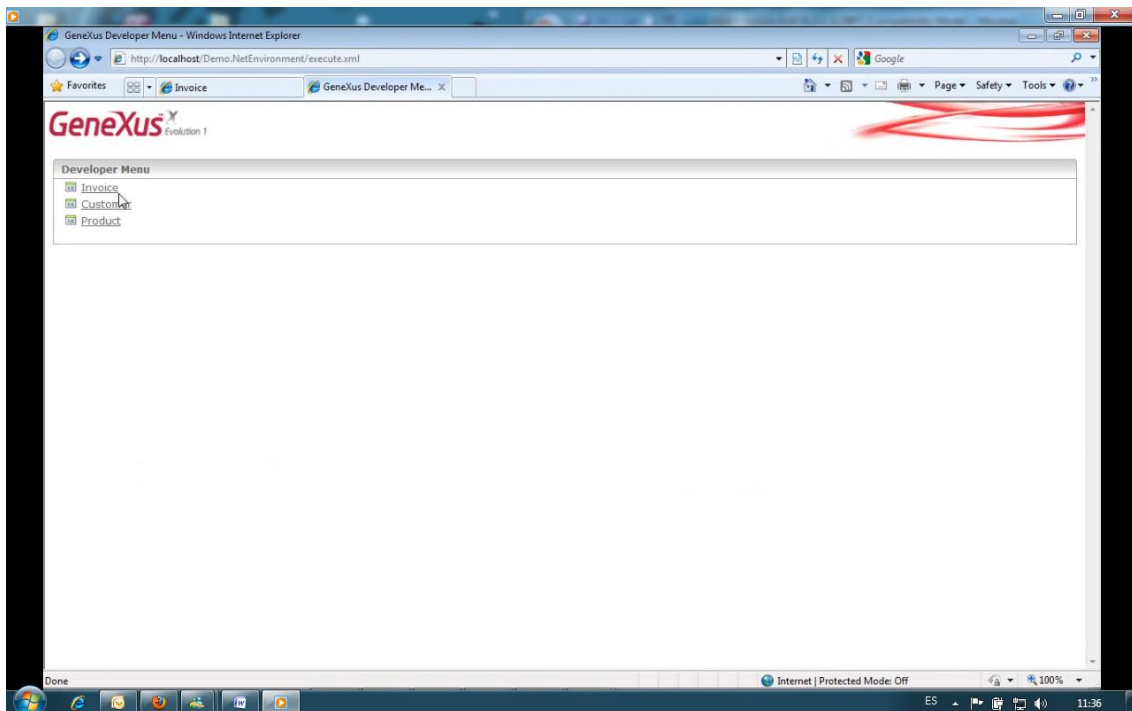


Y ahora veamos algo interesante, recuerden que este producto fue usado para generar una factura. Qué **pasa** si yo lo elimino.

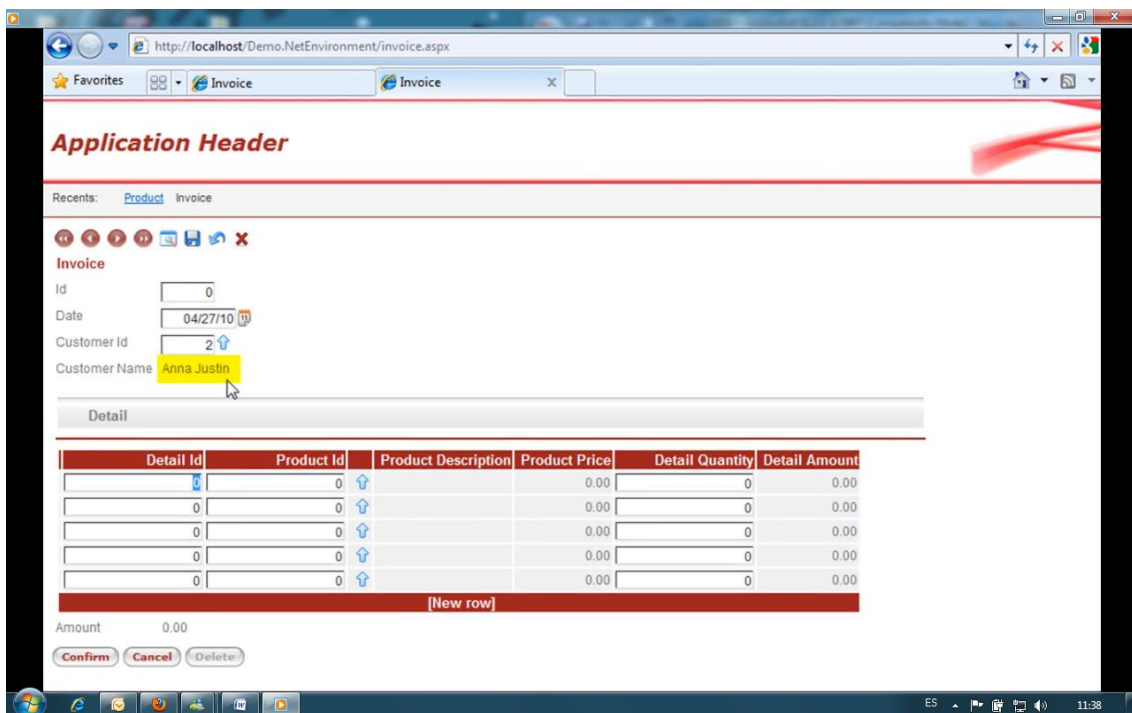


Genexus **no me deja** eliminarlo. Me da un mensaje de error, y yo no programe nada de **eso** porque Genexus también realiza los controles de integridad referencial en transacciones en forma totalmente automática asegurando la integridad de los datos.

Vamos ahora a la transacción de factura para ver **qué cambios** se han **producido** en la factura con todo lo que hemos hecho.



Vamos a agregar una factura nueva.



Observemos que apenas puse el **código** de cliente, Genexus me **recuperó** el nombre del cliente. Esto se hizo sin refrescar la página mediante una combinación de **web services** con **java script**, recuperó la información del servidor y me lo mostró en la página sin que la misma se refrescara.

Application Header

Recents: [Product](#) [Invoice](#)

Invoice

Id:

Date:

Customer Id:

Customer Name: Anna Justin

Detail

	Detail Id	Product Id	Product Description	Product Price	Detail Quantity	Detail Amount
x	1	2	pc	500.00	1	500.00
	0	0		0.00	0	0.00
	0	0		0.00	0	0.00
	0	0		0.00	0	0.00
	0	0		0.00	0	0.00
[New row]						

Amount: 0.00

[Confirm](#) [Cancel](#) [Delete](#)

Lo mismo pasa aquí con la descripción del producto y el precio del producto, que ahora son atributos **inferidos** que no están más almacenados en la tabla de facturas sino que se recuperan de la tabla de productos.

Application Header

Recents: [Product](#) [Invoice](#)

Invoice

Id:

Date:

Customer Id:

Customer Name: Anna Justin

Detail

	Detail Id	Product Id	Product Description	Product Price	Detail Quantity	Detail Amount
x	1	2	pc	500.00	1	500.00
	0	0		0.00	0	0.00
	0	0		0.00	0	0.00
	0	0		0.00	0	0.00
	0	0		0.00	0	0.00
	0	0		0.00	0	0.00
[New row]						

Amount: 500.00

[Confirm](#) [Cancel](#) [Delete](#)

Footer Info

En el próximo **video** veremos una serie de herramientas que aumentan **significativamente** la productividad de su desarrollo.