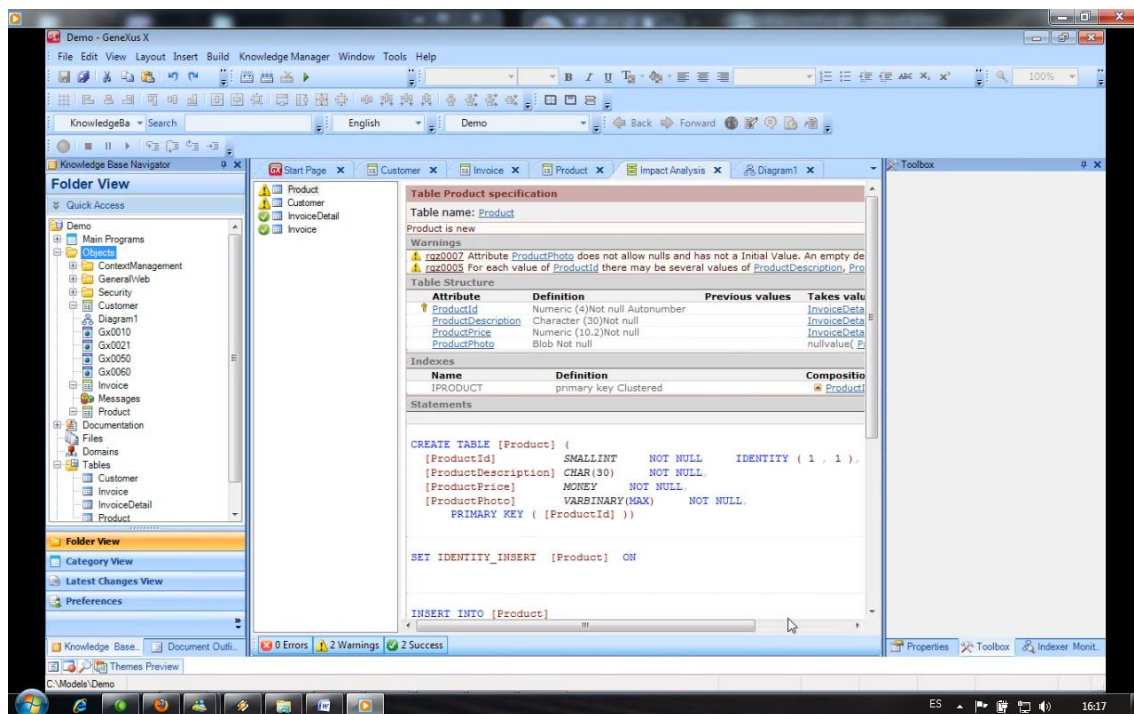


Demo: 4. Cómo obtener pantallas vistosas para trabajar con la información de clientes, productos y facturas.

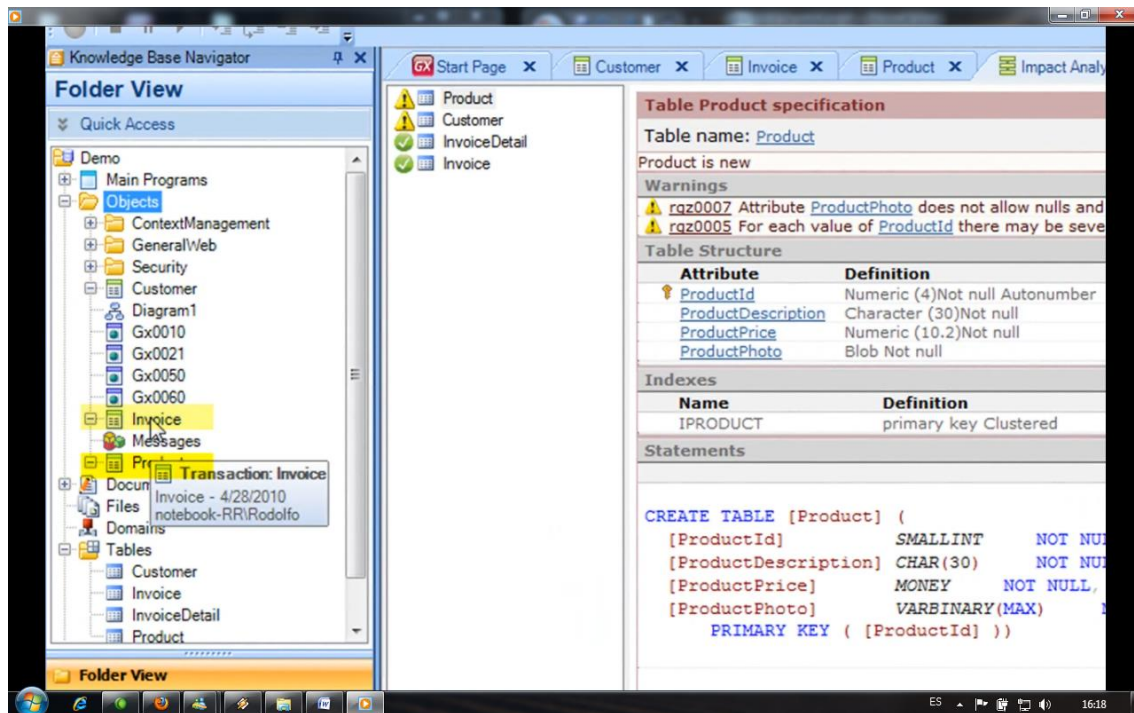
Si tomamos en cuenta lo hecho hasta ahora vemos que si bien el sistema de facturación **cumple** con el cometido de dar **altas, bajas y modificaciones** en la base de datos, sería cómodo que pudiéramos trabajar con una lista de **facturas, clientes o productos** y navegar a través de ellos.

Por ejemplo, en el caso de clientes que podamos filtrar la lista por cierto **criterio**, editar la información de un **cliente**, crear uno **nuevo** o que se puedan ver las facturas de un cliente o datos relacionados.

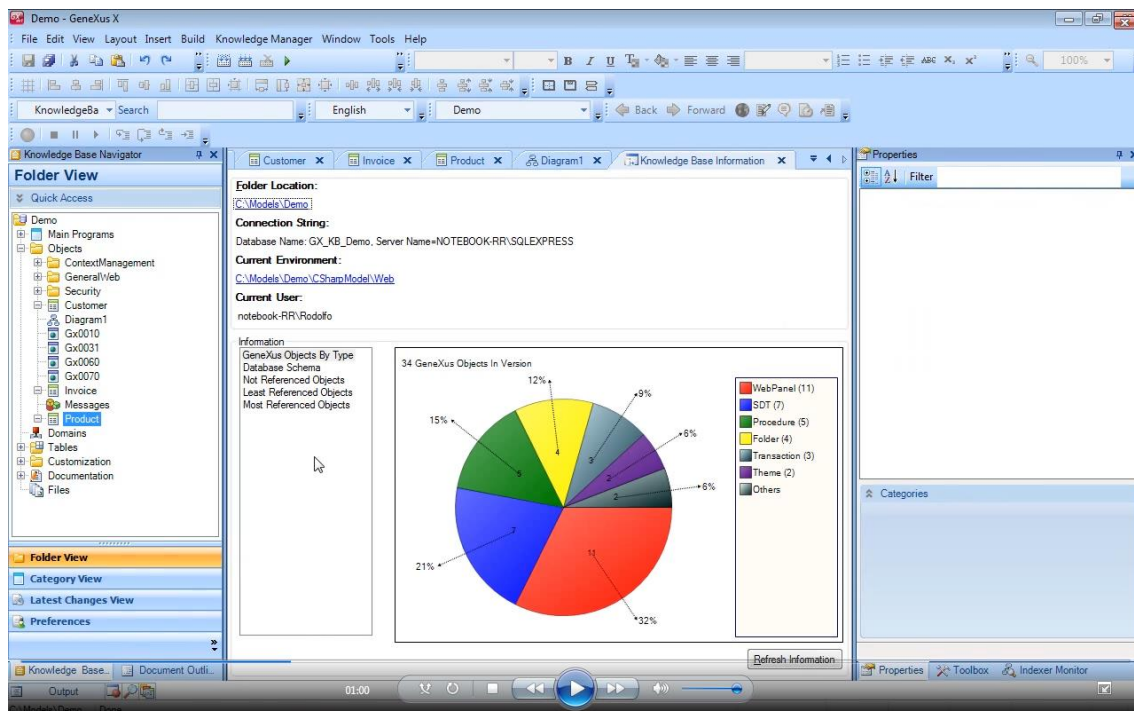
Y vemos que si trabajamos con clientes, productos o facturas esa **lista** de selección son todas parecidas, es decir, que siguen cierto **patrón**.



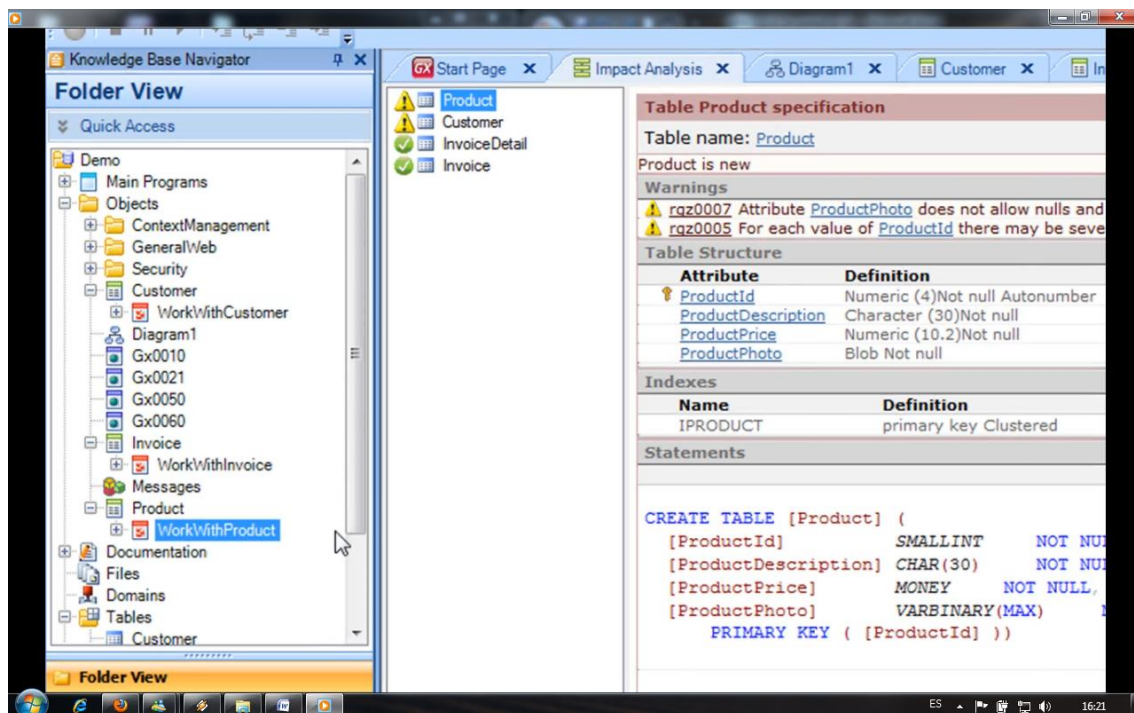
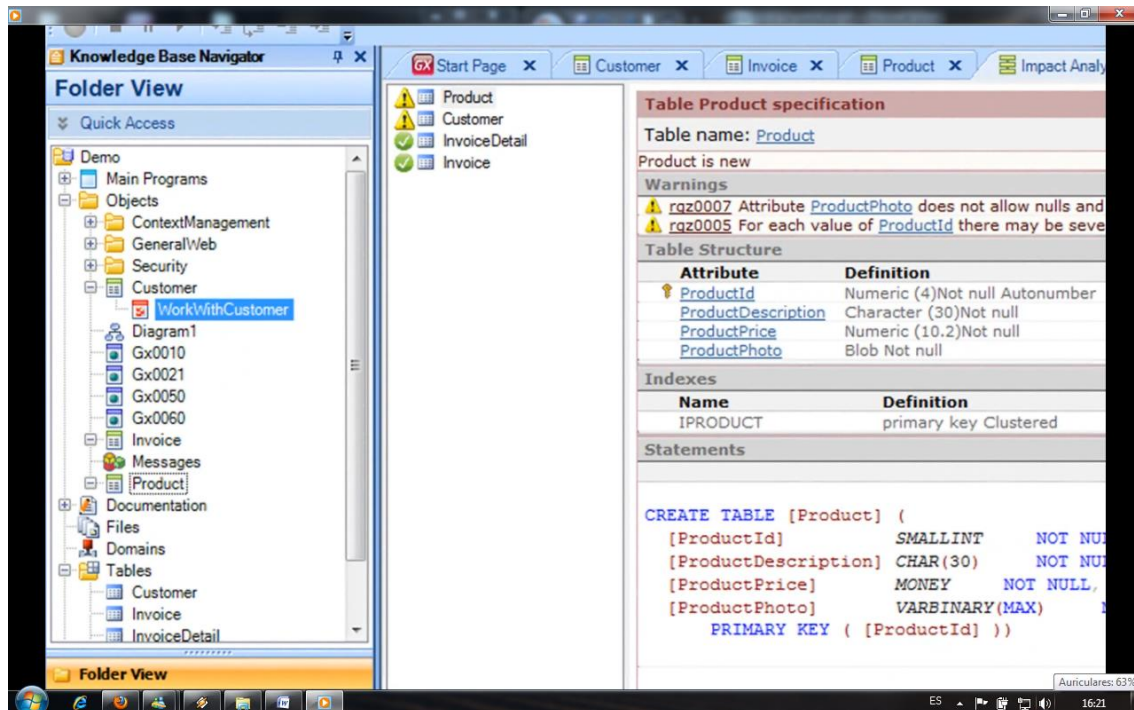
GeneXus nos provee un mecanismo para aplicar patrones y agregar cierta funcionalidad a la aplicación. Al aplicar el **patrón**, GeneXus **crea** automáticamente en la KB todos los objetos necesarios para agregar dicha funcionalidad.



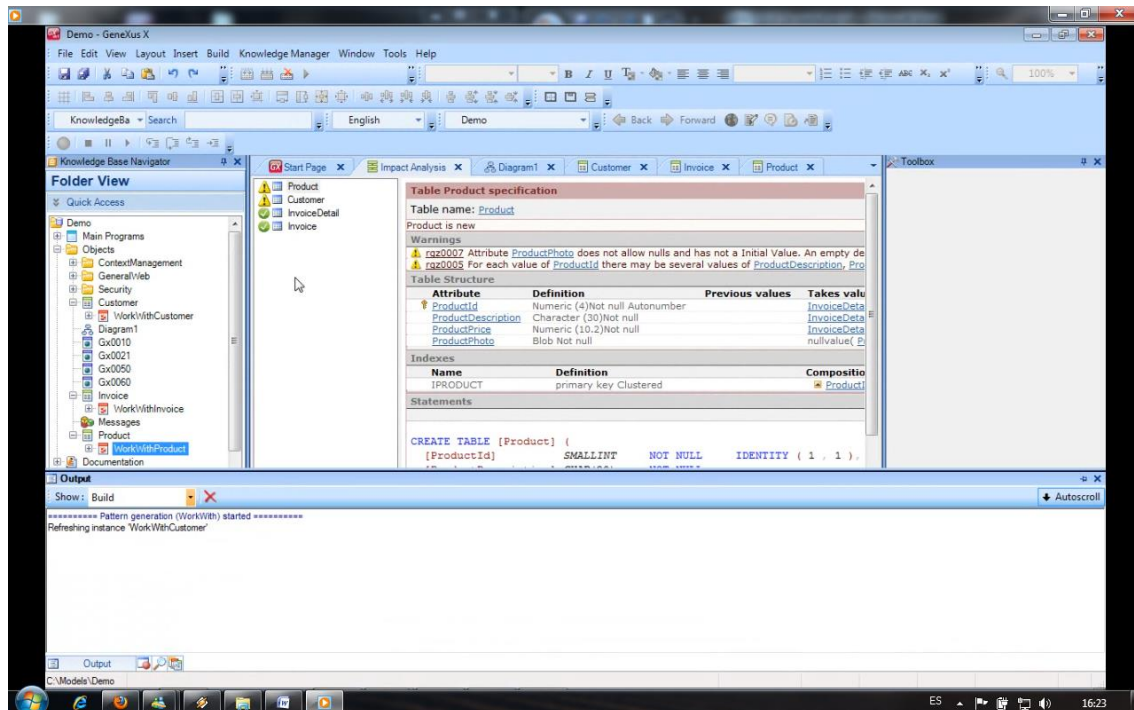
En este caso aplicaremos el patrón WorkWith a las transacciones de clientes, facturas y productos.



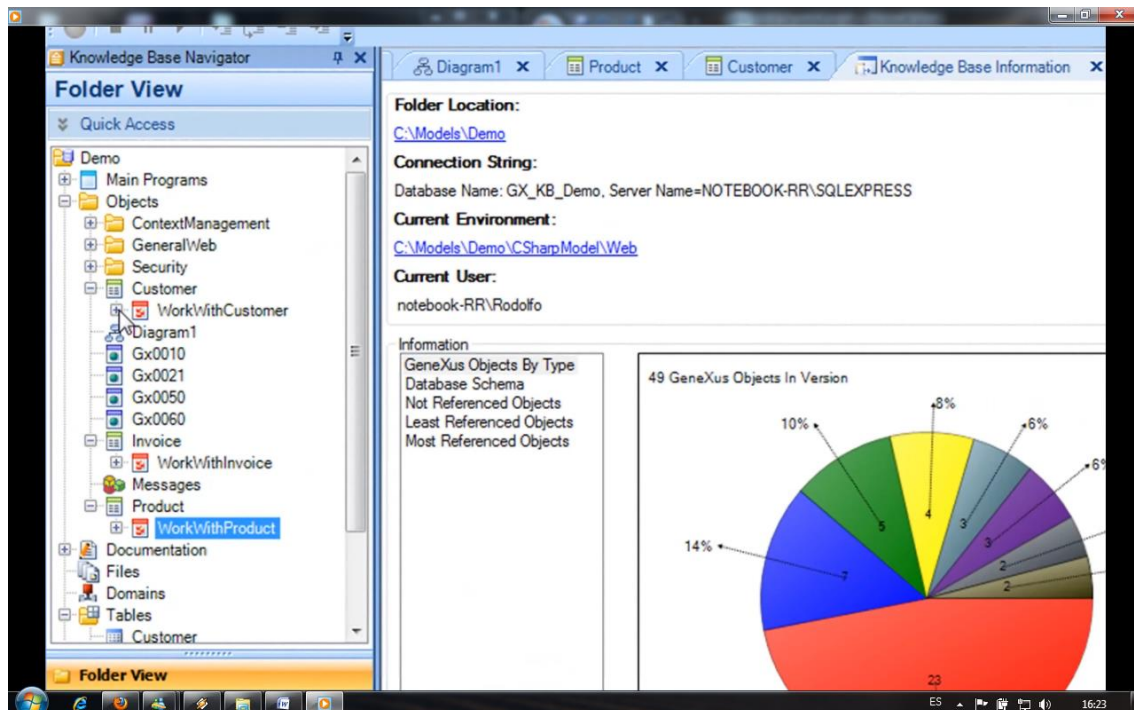
Primero vamos a ver cuántos objetos tenemos ahora en la KB y ahora vamos a aplicar el patrón a estas 3 transacciones.



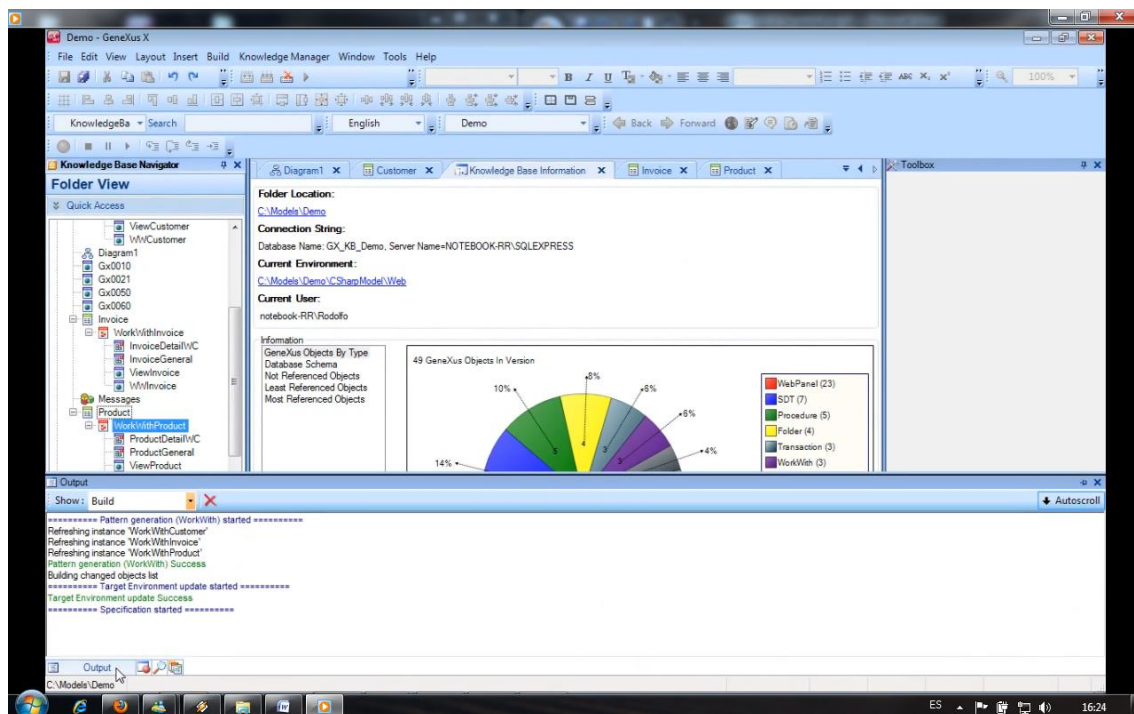
Observamos que bajo **Customer** se me creó un nodo que dice **WorkWithCustomer**, lo mismo bajo **Invoice** y lo mismo va a pasar debajo de **Product**.



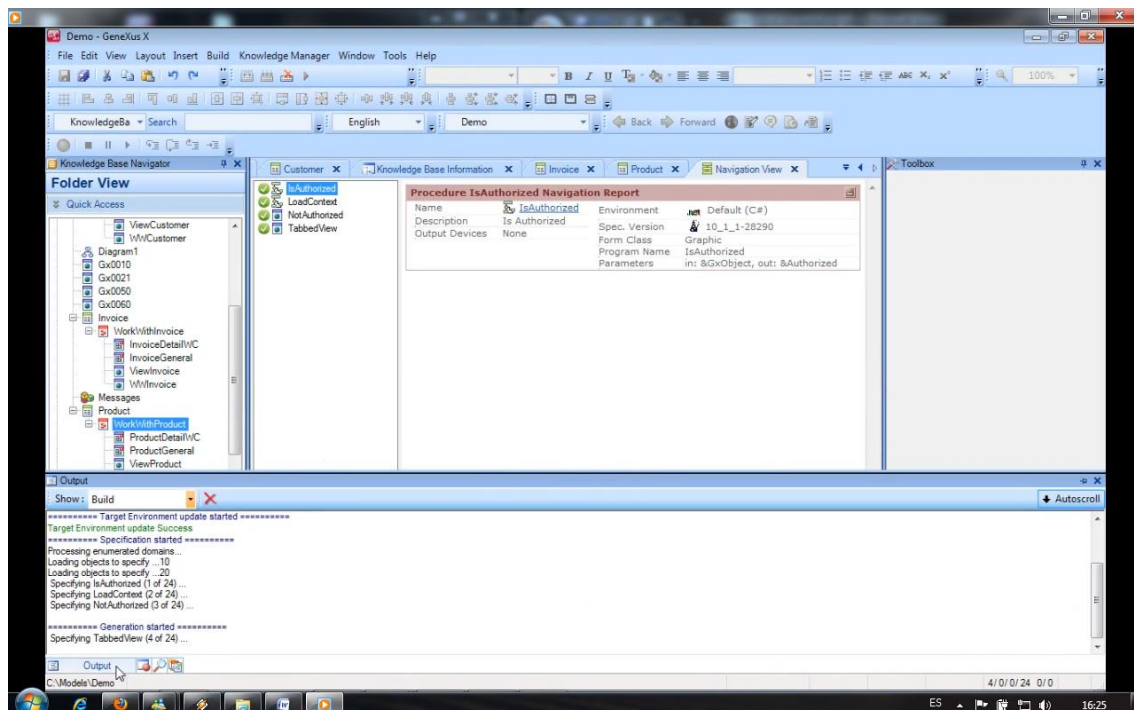
Vamos a dar a **ejecutar**, mientras tanto podemos ver la cantidad de objetos que tenemos ahora creados en la KB.



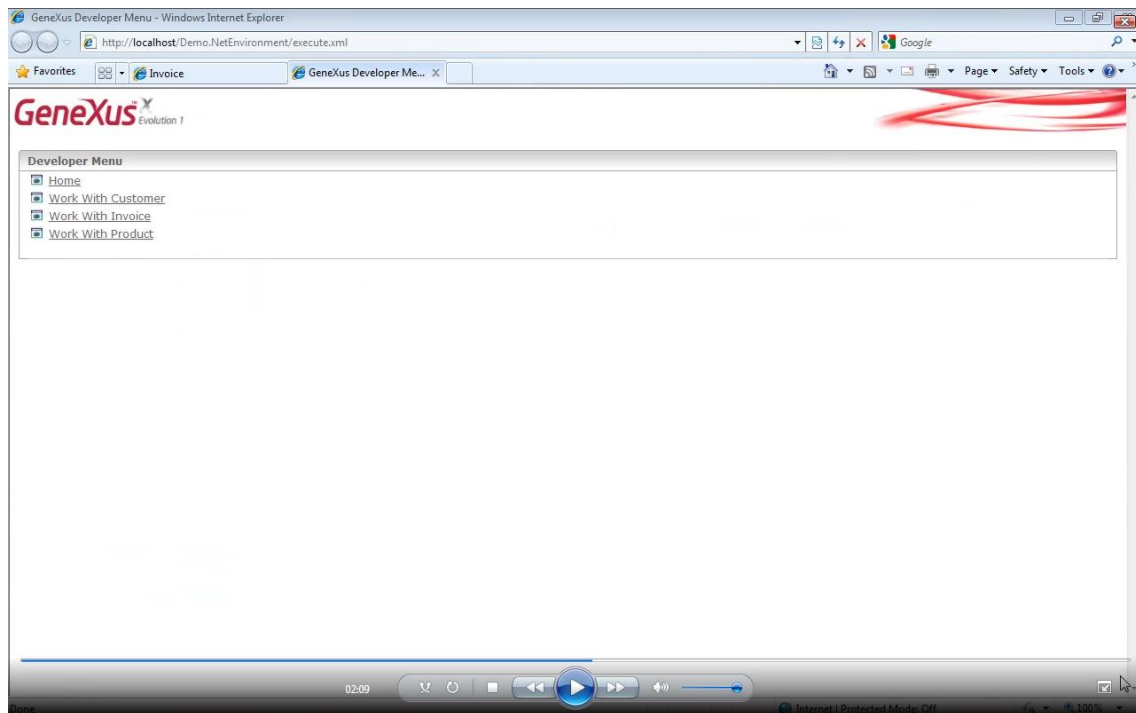
Son todos los objetos que se crearon debajo de cada uno de estos nodos que hacíamos referencia hoy.



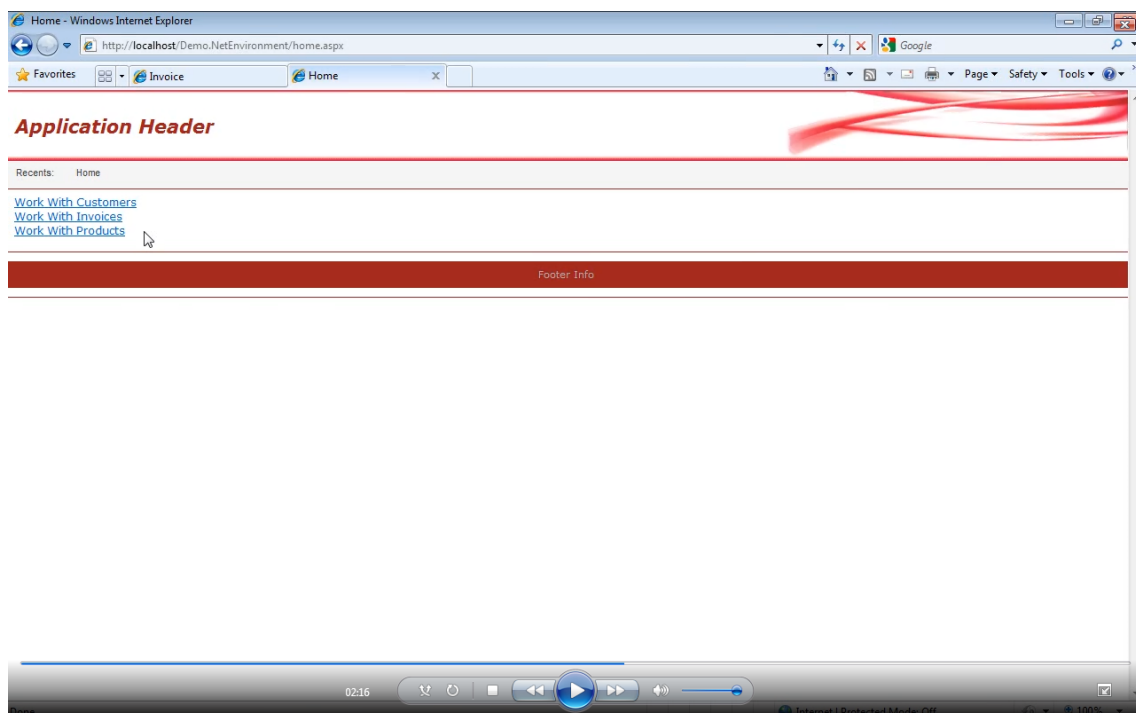
En este momento GeneXus está creando todo lo necesario, es decir, objetos relacionados con la seguridad,



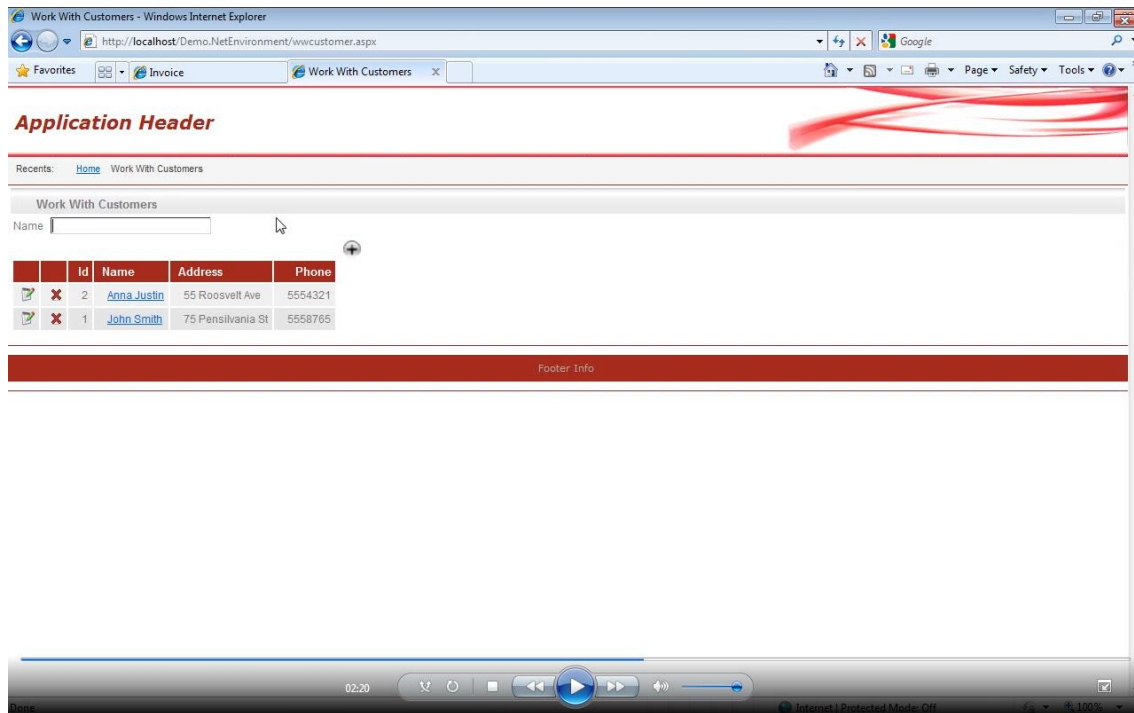
objetos para acceder a esa lista de selección que mencionábamos previamente de forma tal de que nos resulte después mucho más sencillo poder encontrar la información relacionada con cada uno de los clientes, productos o facturas. Compilando en estos momentos la aplicación.



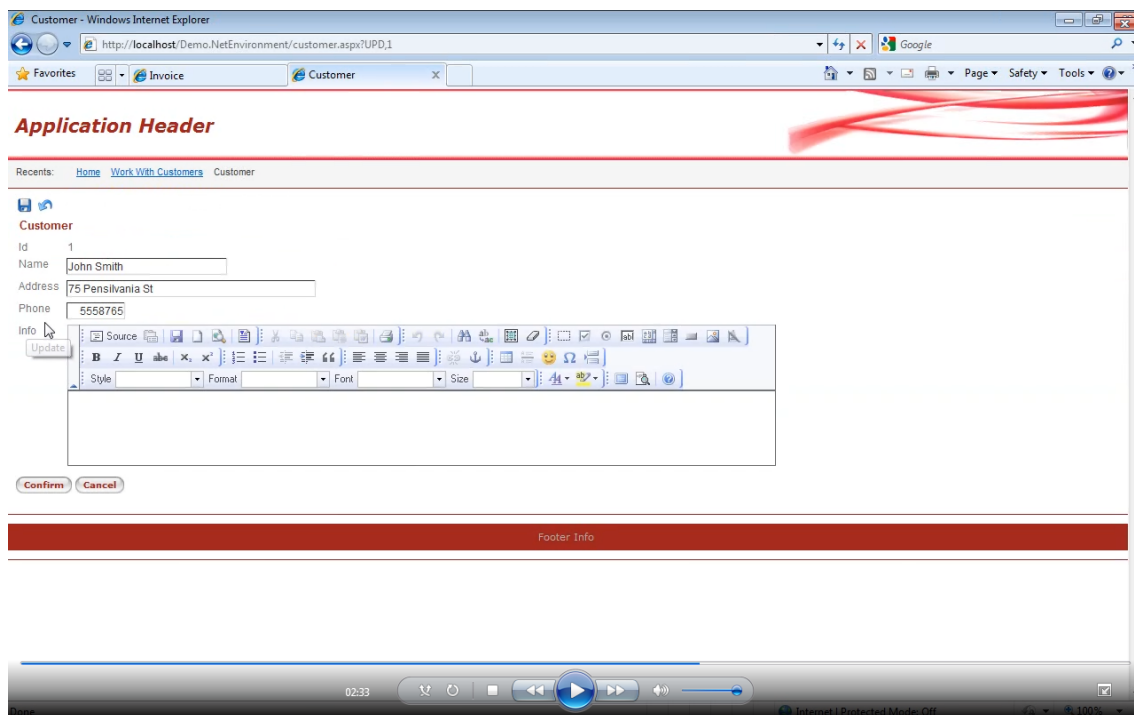
Y aquí vemos a la aplicación en **ejecución**. GeneXus me agrega un **link**



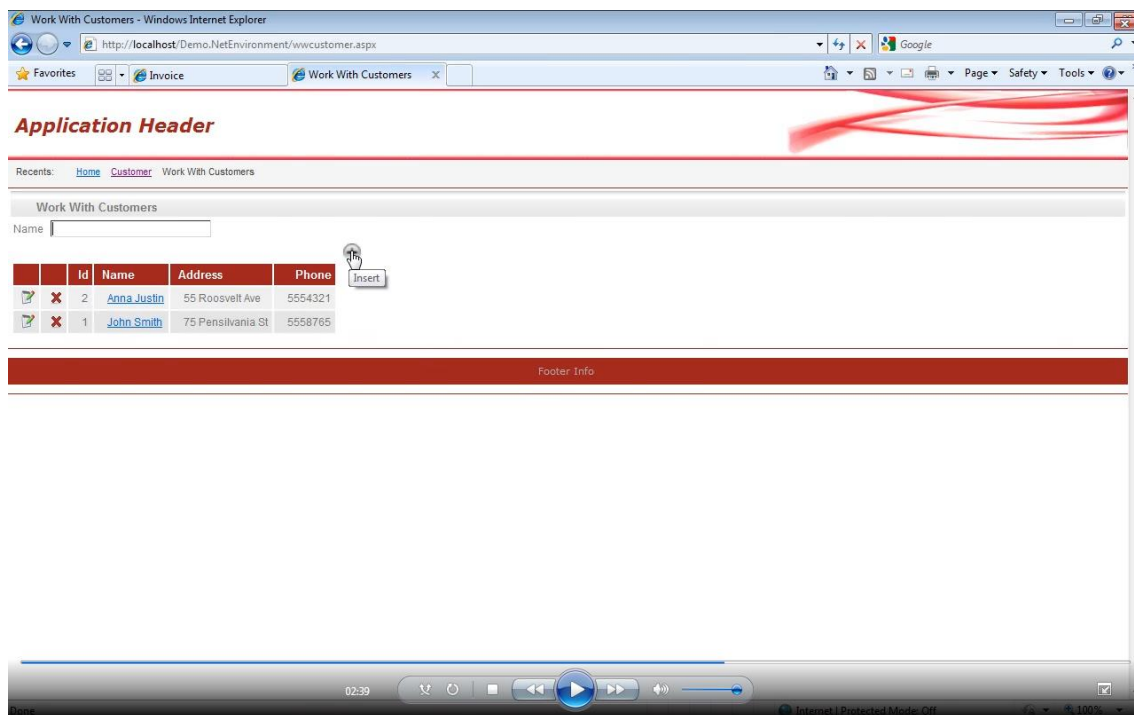
mediante el cual puedo acceder a los distintos trabajar con, por ejemplo, trabajar con clientes.



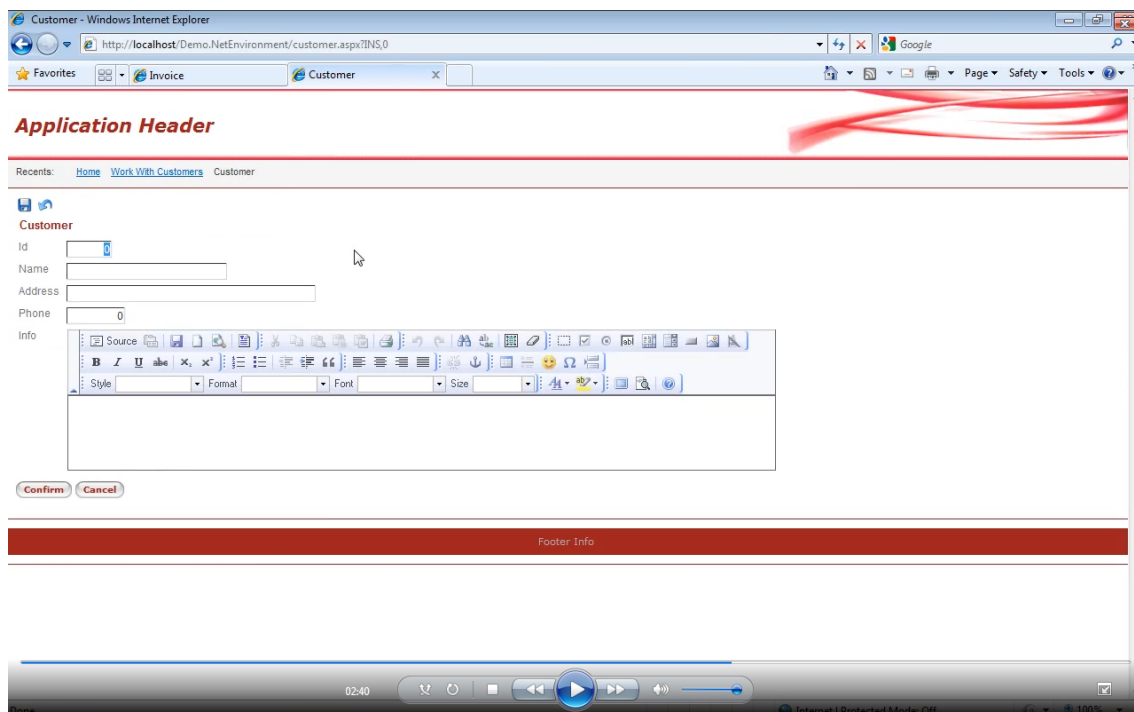
Aquí vemos en una **lista** todos los clientes que tenemos **definidos**, podemos inclusive poner **filtros**, podemos



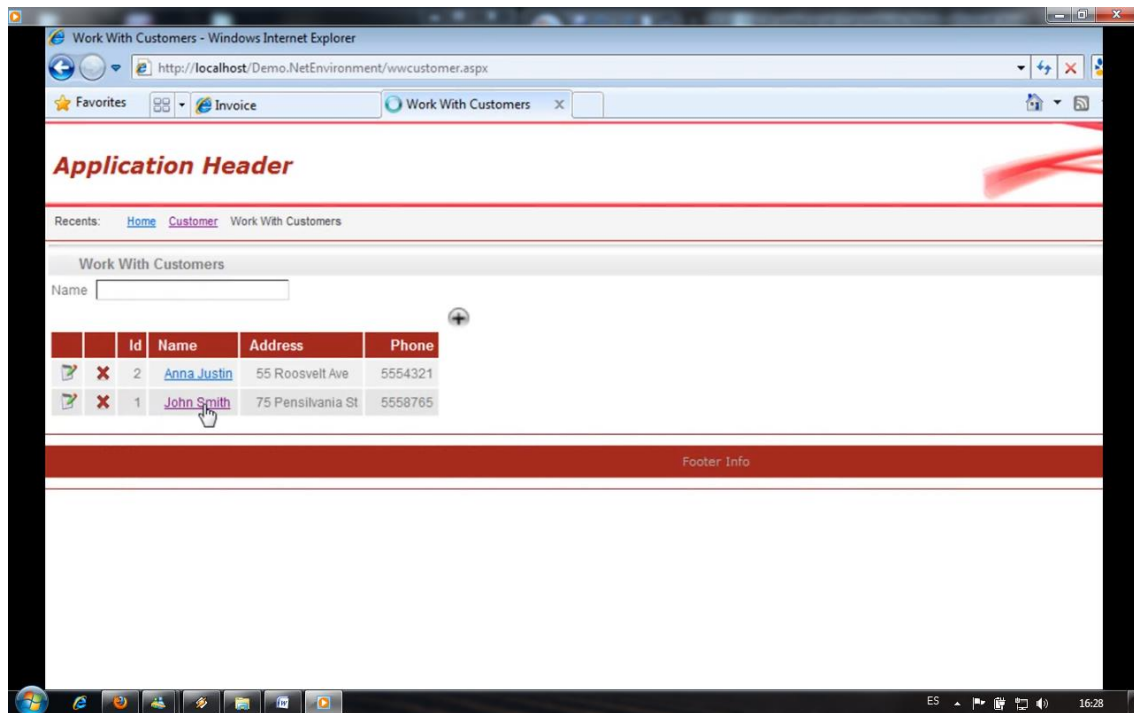
editar la información de un cliente.



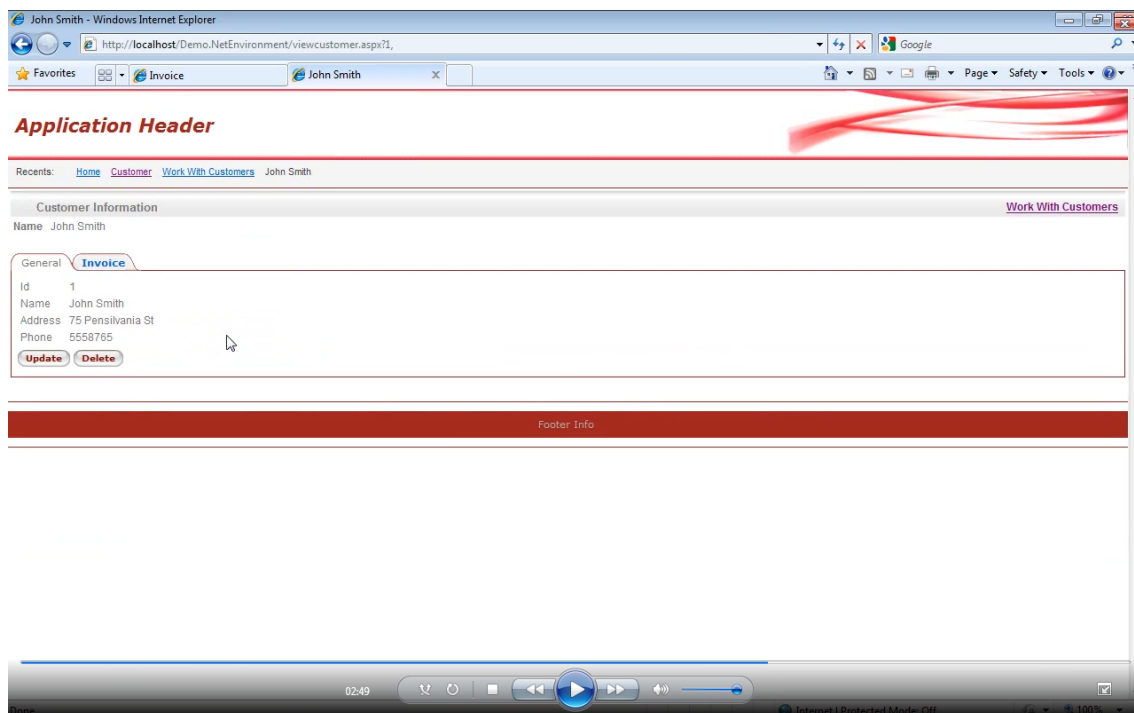
Podemos eliminar un cliente o inclusive ingresar



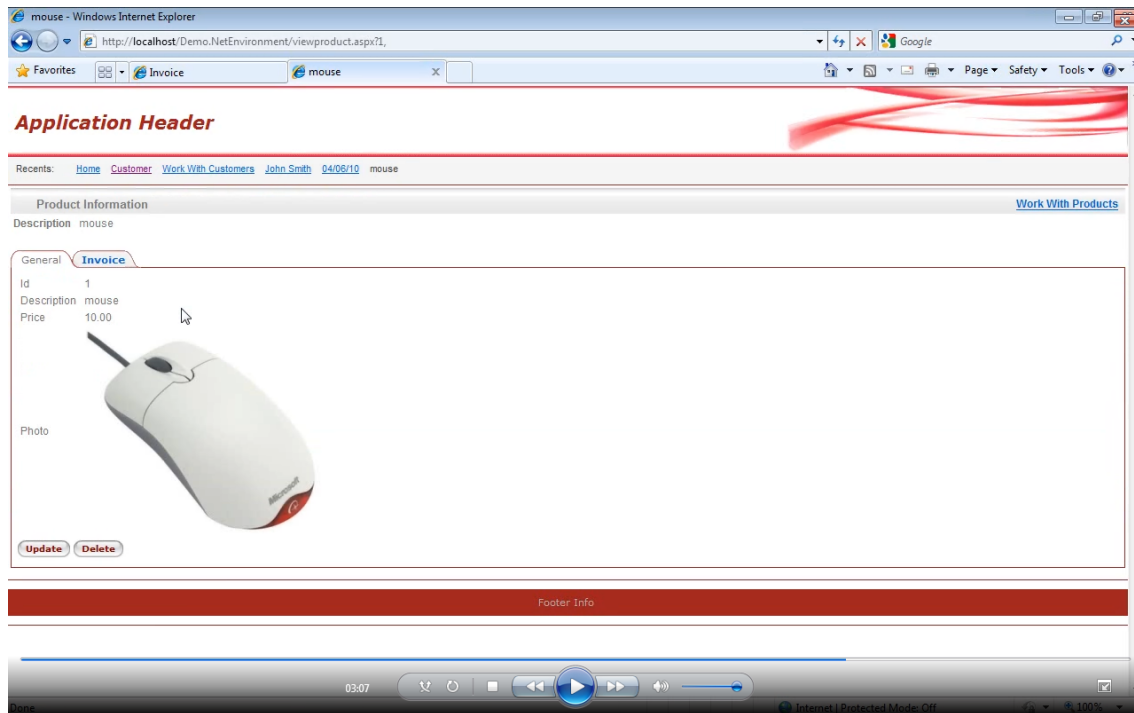
un cliente nuevo.



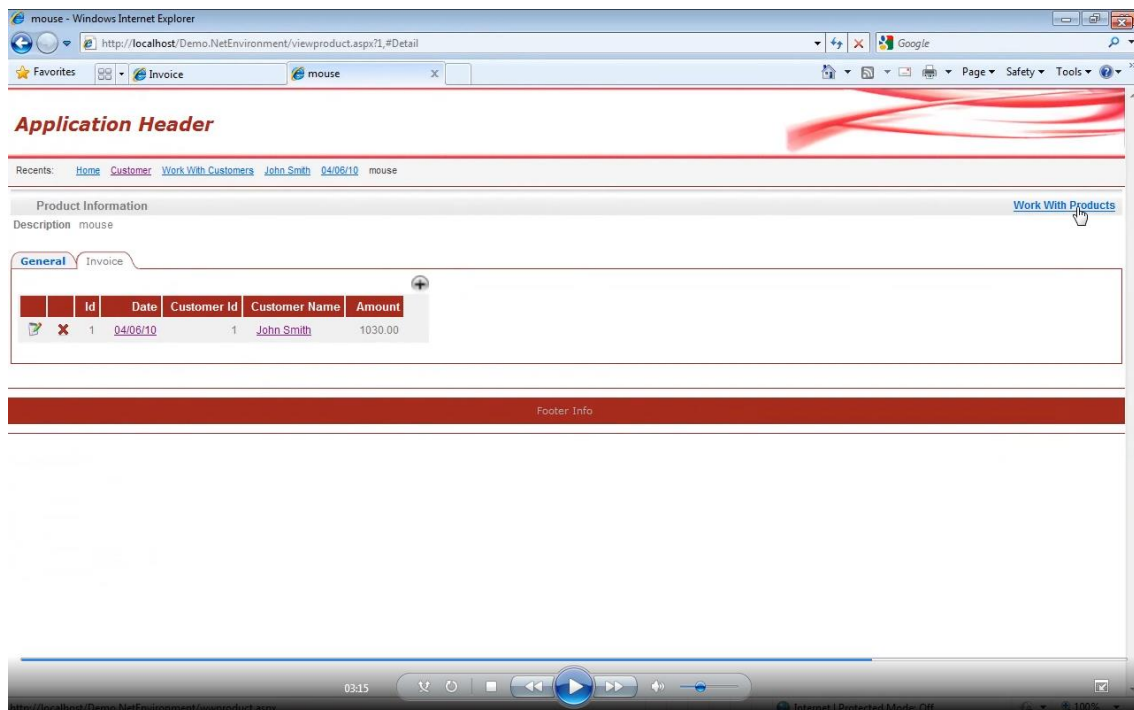
Todo a partir de esta pantalla. Dado también un cliente tenemos un link donde



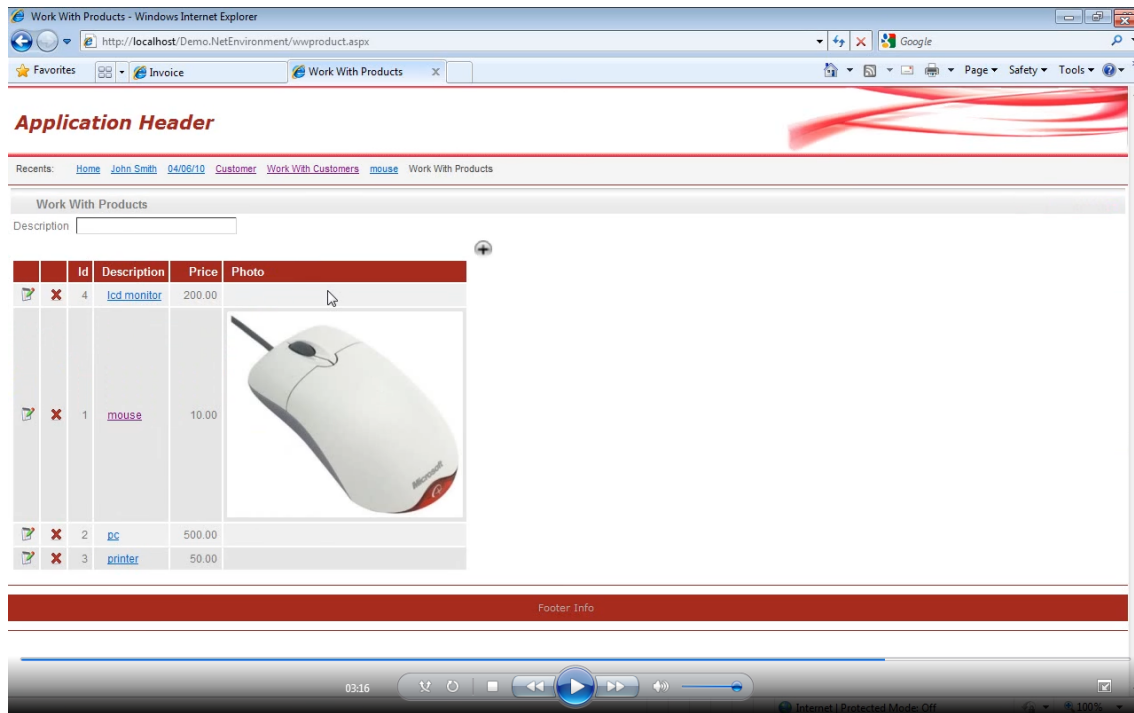
podemos ver su información y toda la información relacionada al mismo, por ejemplo las facturas de ese cliente.



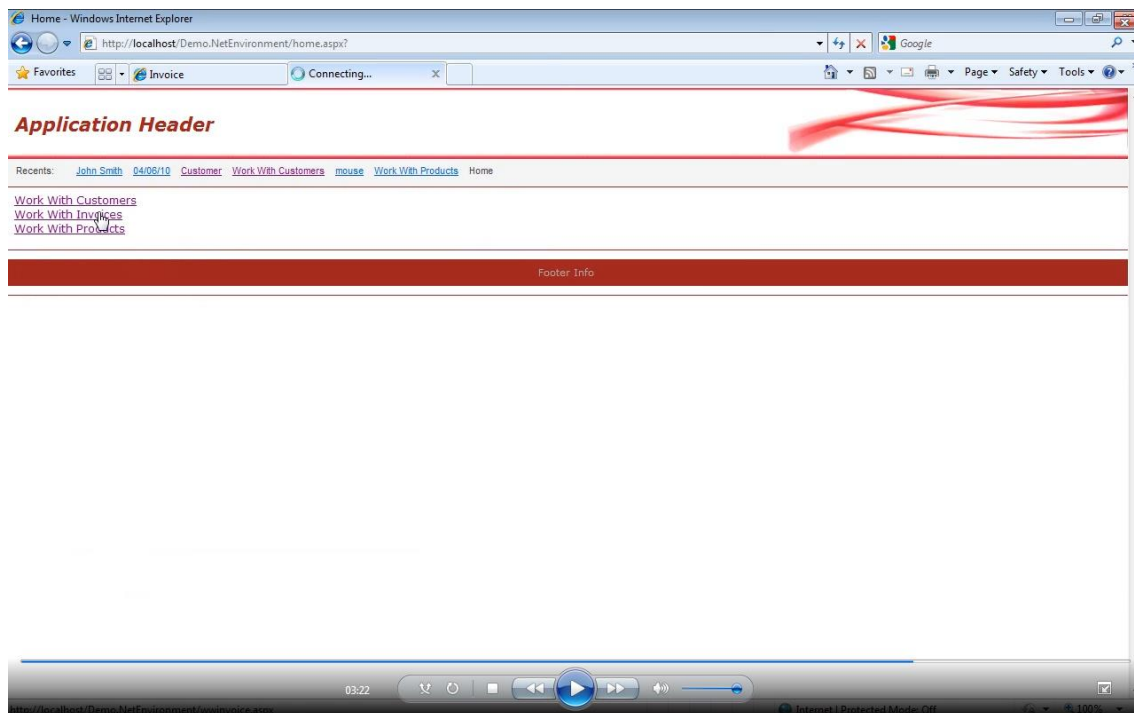
datos de ese producto y en qué **facturas** se



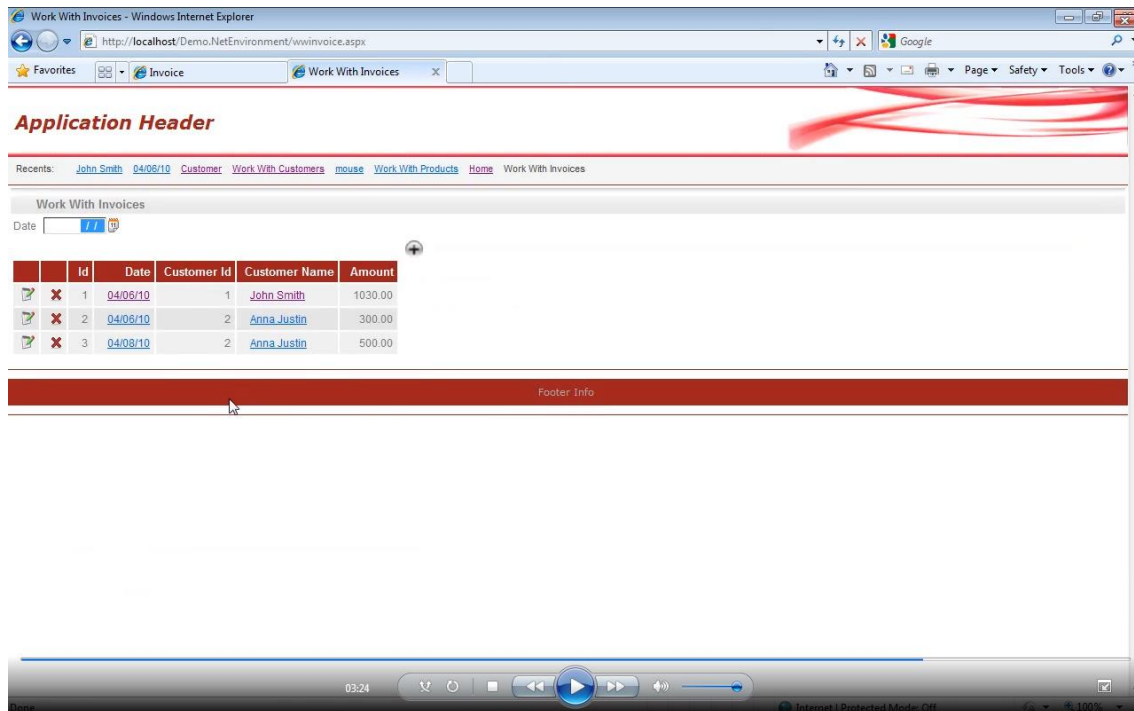
incluye ese **producto**. Podemos ir también a trabajar con **productos**



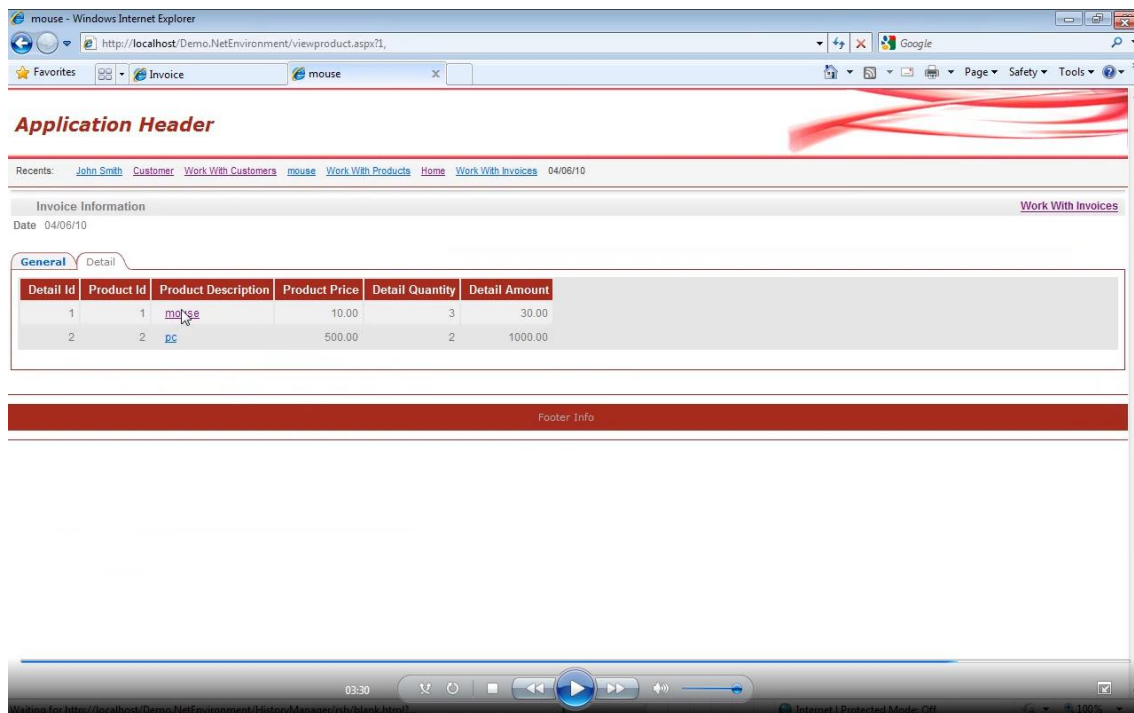
donde tenemos una funcionalidad similar a la que tuvimos con trabajar con **clientes** o



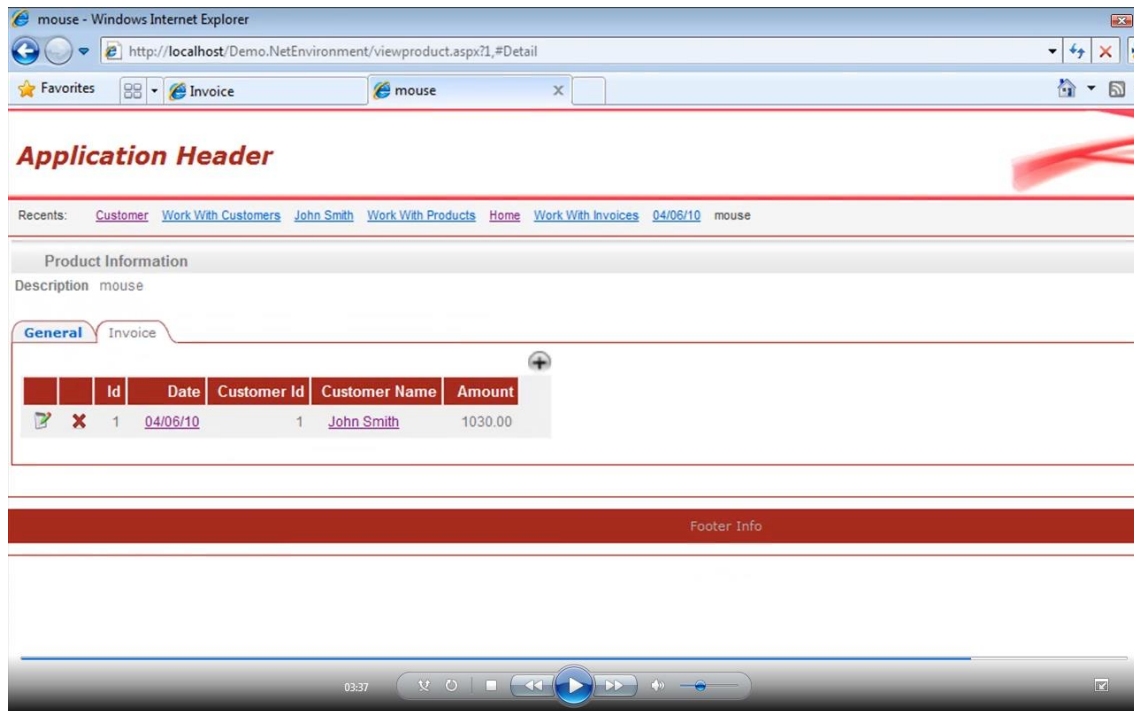
podemos ir también a trabajar con facturas.



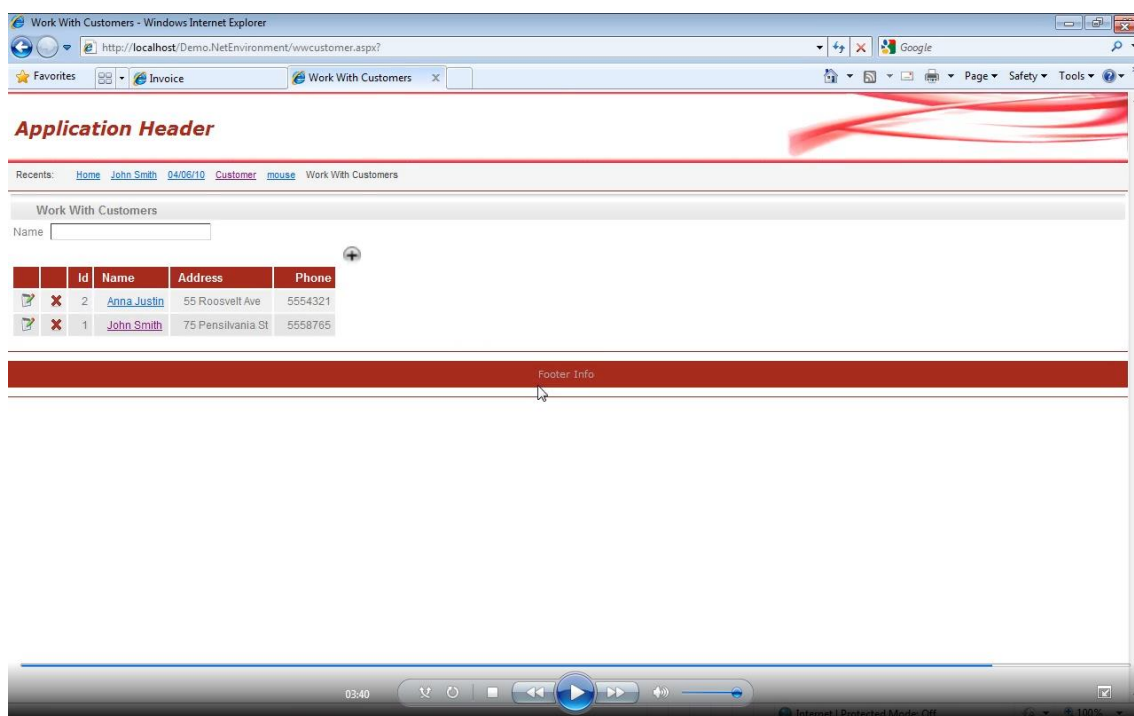
Donde podemos ver todas las facturas, a que clientes pertenecen y



así podemos seguir mostrando más datos.

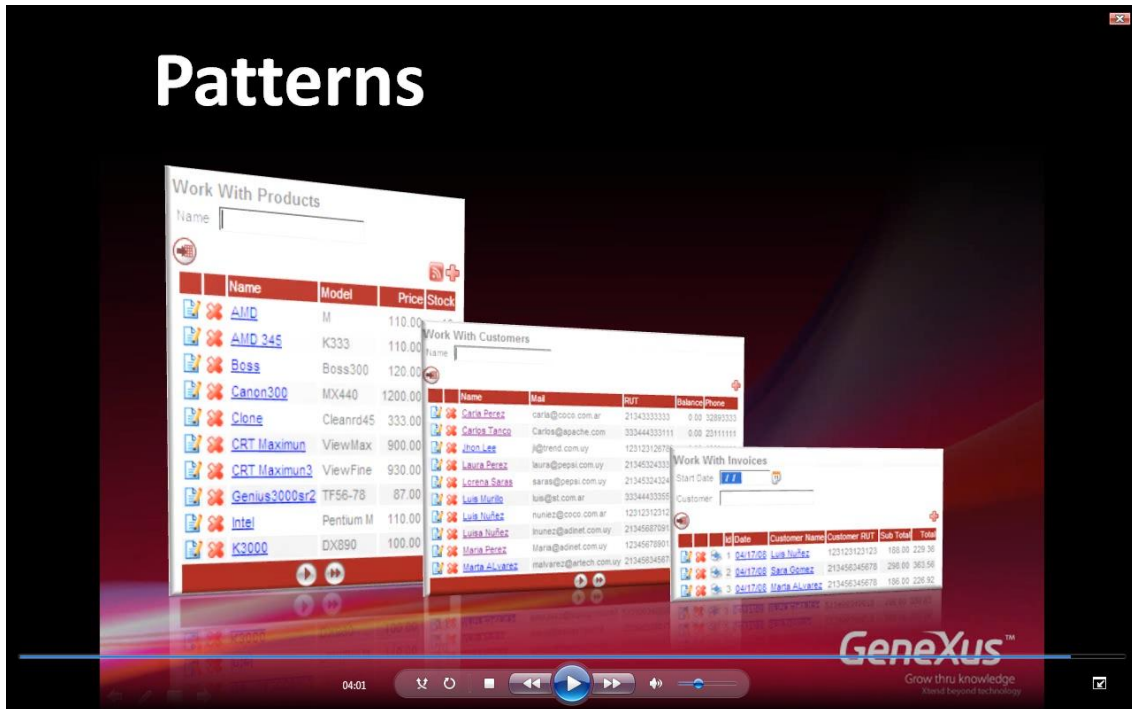


Quiere decir que dado un **concepto** vemos toda esa información relacionada. Esto qué significa, que GeneXus cuando creó los **objetos** en la KB,



creó una serie de objetos que están relacionados entre sí y nosotros no tuvimos que decirle cual era dicha relación. Esta es la ventaja de trabajar con una herramienta inteligente como **GeneXus** que obtiene sus propias inferencias a partir de una base de conocimiento.

Patterns



Hemos visto como mediante la aplicación de **patrones** podemos mejorar nuestra **aplicación** con muy poco esfuerzo. En esta misma línea vamos a volver al **IDE** para crear una galería de imágenes.