

Ejecución de la aplicación por primera vez

GeneXus[™]

Modos de Prototipado

Versión Full

- Local
- Nube
- Cualquier lenguaje

Prototipado

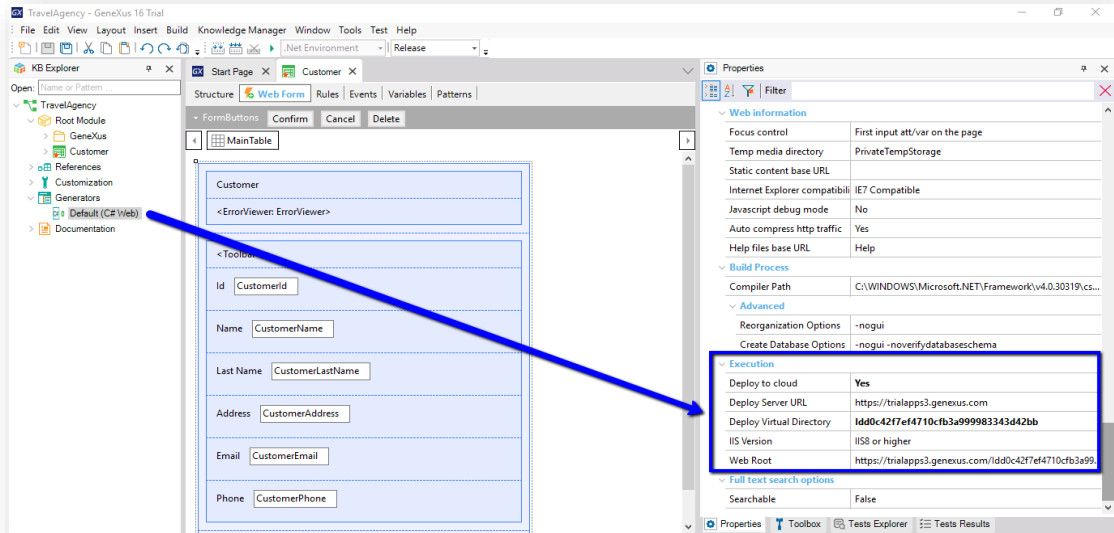
Versión Trial

- Nube
- .Net

Queremos probar la transacción que creamos en los pasos anteriores. Para ello, le vamos a pedir a GeneXus que construya la aplicación, que cree la base de datos, que haga todo lo que tenga que hacer para tener los programas ejecutables para que podamos probar. A ese ciclo de pruebas en el que un desarrollador se embarca antes de tener la aplicación lista, le llamamos prototipación.

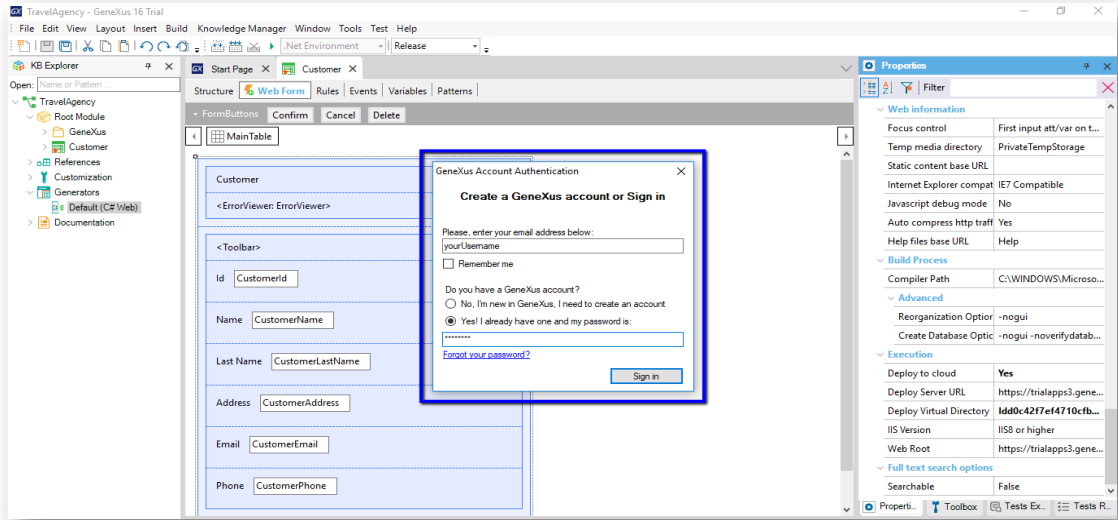
En la versión full de GeneXus se le permitirá prototipar la aplicación tanto en forma local (es decir, los programas y la base de datos estarán en servidores de su red) como en una nube que GeneXus tiene a disposición. Además, podremos elegir hacerlo en cualquiera de los lenguajes de programación disponibles.

Modo de Prototipado: en la Nube



La versión Trial, en cambio, sólo ofrece prototipar en la nube y en lenguaje .Net. Lo vemos aquí, abrimos la ventana de Propiedades y acá observamos cómo solamente podemos prototipar en la nube.

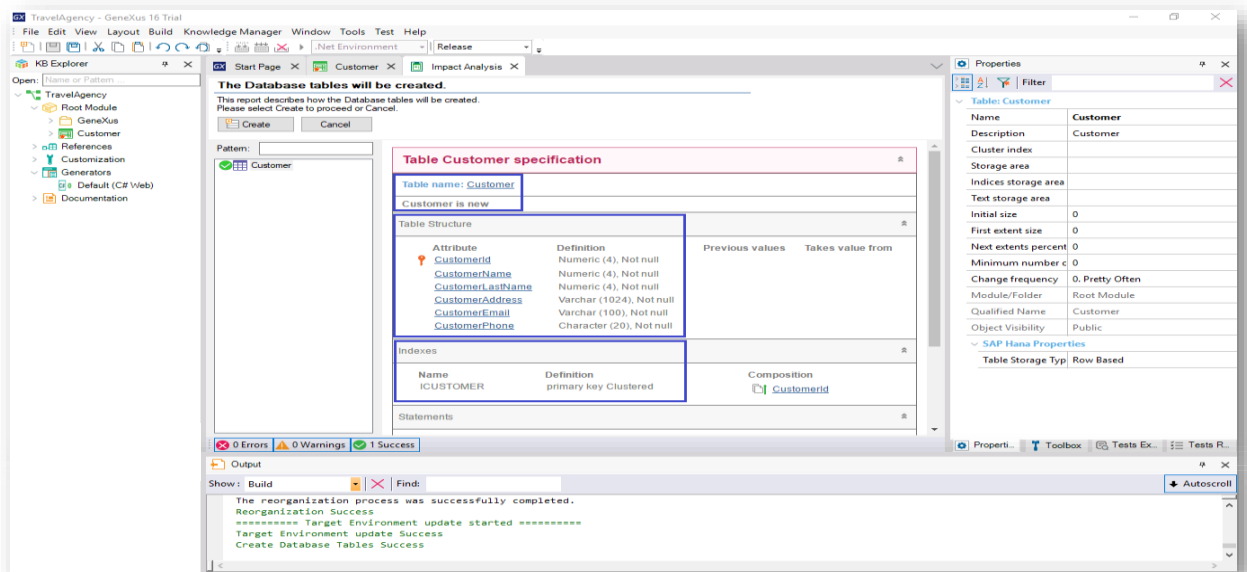
Modo de Prototipado: en la Nube



Es lo que haremos ahora mismo, presionando la tecla de función F5.

Nos pide, por ser la primera ejecución, que nos logueemos usando nuestro usuario GeneXus. Si aún no tenemos uno, nos ofrece crearlo.

Análisis de Impacto



Tras loguearnos GeneXus comienza a trabajar, concluyendo con un informe que recibe el nombre "Impact Analysis", Análisis de Impacto.

En dicho informe GeneXus analiza el impacto causado por las nuevas definiciones hechas en la base de conocimiento y nos informa qué creaciones o cambios estructurales detecta que debe realizar en la base de datos.

Si miramos el informe en detalle... observemos que nos está avisando que hay una tabla nueva de nombre Customer para ser creada...

y que la estructura de dicha tabla contendrá los siguientes atributos:

Vemos que GeneXus también creará automáticamente un índice por CustomerId; más adelante explicaremos este concepto.

Dado que estamos de acuerdo con que se cree esta tabla con esta definición en la base de datos, presionamos el botón CREATE, y GeneXus procede a crear los programas necesarios para crear la base de datos (ya que no existía) y la tabla Customer, con esta estructura en dicha base de datos.

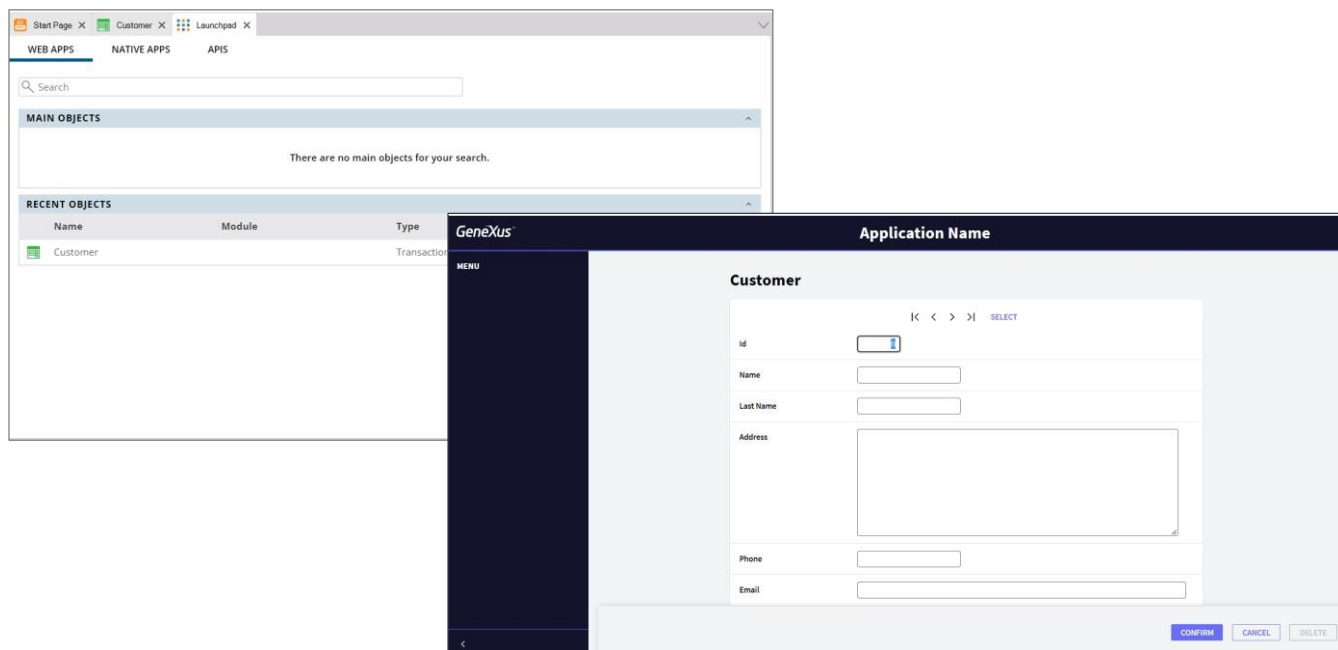
A continuación GeneXus ejecuta estos programas, y queda creada la base de datos. GeneXus continúa generando otros programas (es decir todas las líneas de código necesarias en el lenguaje de programación elegido para que nuestra aplicación haga lo que deseamos, en este caso ingresar datos de clientes)... y nos muestra si el resultado fue exitoso, o surgió algún error o advertencia.

Ejecutando la aplicación

- En GeneXus, cree todo lo que se ha analizado hasta ahora.
- Ejecute la aplicación por primera vez (F5).
 1. Análisis de Impacto: GeneXus analiza el impacto causado por nuestras definiciones e informa sobre cualquier creación o cambios en la estructura detectados en la BD.
 2. Si estamos de acuerdo y procedemos → GeneXus crea automáticamente:
 - la Base de Datos
 - la tabla Customer
 - el programa asociado a la transacción Customer + programas que proporcionan funcionalidades complementarias
- Realizar el ingreso, modificación y eliminación de clientes.



Ejecutando la aplicación



A continuación, aparece este launchpad que permite seleccionar el objeto a ejecutar, así que seleccionamos a la transacción Customer

Y tenemos la aplicación en funcionamiento:

Aquí estamos viendo una página que nos permite ingresar, modificar y eliminar clientes. Este menú que vemos aquí ofrece acceso a las transacciones y a otros objetos que veremos más adelante.

Ingresemos los datos de nuestro primer cliente.

Este cliente se identificará con el número 1.

Se llamará John. Su apellido es Smith. Vive en la quinta avenida. Su teléfono es: 1111, y su correo electrónico es jsmith@hotmail.com.

Ahora pulsamos el botón Confirm y un mensaje nos avisa que los datos han sido agregados satisfactoriamente, mientras que el formulario se limpia y queda listo para un nuevo ingreso.

Ingresemos un 2do cliente... lo identificamos con el número 2:

Se llama Susan. Su apellido es "Brown". Su dirección es en la séptima avenida. Su teléfono es 2222, y su correo es sbrown@gmail.com.

Seleccionamos el botón Confirm, y nuevamente sale el mensaje que avisa que los datos han sido agregados satisfactoriamente:

Vamos a buscar los datos de John, digitamos su número de cliente (el uno) y luego presionamos tabulador.

Vemos que se muestran todos los datos de John y una que vez que estamos posicionados aquí podríamos cambiar algún dato, por ejemplo su teléfono.

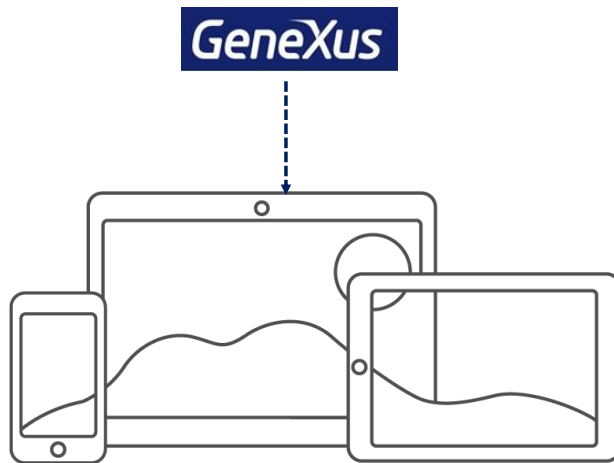
O podríamos eventualmente eliminar el cliente John del registro de clientes con el botón Delete.

También podemos usar los botones de aquí arriba, para navegar entre clientes, yendo por ejemplo al siguiente, al anterior, al primero o al último, o para buscar a un cliente en particular mediante la ventana de selección. Esa ventana fue creada automáticamente por GeneXus y corresponde al objeto Web Panel que se muestra.

Observemos que si ajustamos el tamaño de la pantalla, los controles se ajustan al espacio disponible.

Aplicación Web Responsive

- Brinda una visualización óptima, una experiencia de navegación muy buena y un excelente uso de los espacios de la pantalla, independientemente del tamaño de la misma.

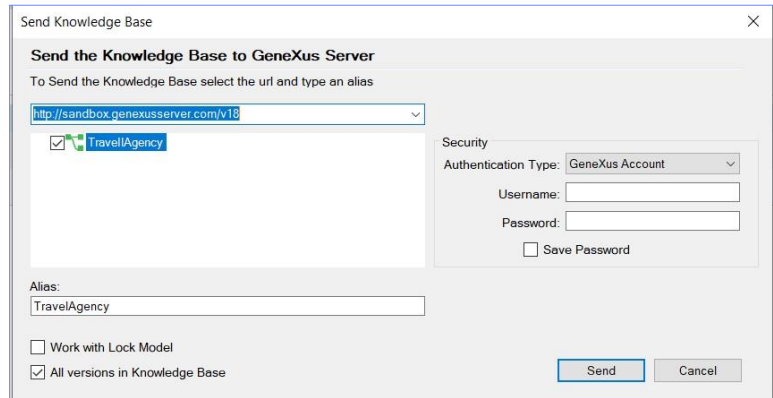
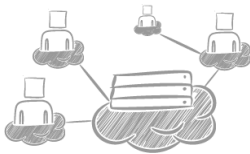
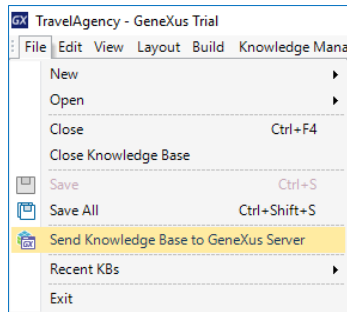


¿Qué significa esto? GeneXus genera aplicaciones web responsive, o aplicaciones sensibles, que ofrecen una óptima visualización, buena navegación, y un excelente aprovechamiento de los espacios de la pantalla.

Todo esto se logra utilizando un diseño que permite ver toda la información de una forma armoniosa dentro de la pantalla, y GeneXus nos provee de funciones de edición para lograrlo. Las mismas nos brindan la posibilidad de mostrar diferentes formas de visualizar el layout dependiendo del tamaño de la pantalla del dispositivo.

Ahora que hemos visto en ejecución todo lo que GeneXus generó automáticamente a partir de nuestra definición de la transacción Customer... volvamos al ambiente de desarrollo.

Send Knowledge Base to GeneXus Server



Ahora vamos a subir esta base de conocimiento a un servidor de GeneXus en la nube. Aunque no es obligatorio, nos conviene utilizar GXserver para que nos vaya quedando registrada la actividad histórica del proyecto, y mucho más, si trabaja más de una persona en la misma base de conocimiento.

GeneXus Server nos permite centralizar nuestra base de conocimiento (KB) y facilitar la gestión si trabajamos en equipo sobre ella, incluso si los usuarios se encuentran en diferentes puntos geográficos. Cada desarrollador trabaja sobre su propia copia local de la KB, la cual a su vez está conectada a la KB centralizada administrada por GXserver.

Tener una KB central administrada por GeneXus Server permite que cualquier desarrollador autorizado, pueda enviar y recibir modificaciones desde su propio lugar de trabajo.

Esto nos permite desarrollar de forma compartida entre varios usuarios trabajando cada desarrollador en forma independiente, pero como si se estuviera trabajando en forma centralizada, ocupándose GeneXus Server de consolidar los cambios realizados al proyecto por cada desarrollador.

Para subir nuestra KB vamos a File y elegimos Send Knowledge Base to GXserver. Vemos que por defecto se nos ofrece la siguiente dirección del servidor donde podemos subir nuestra KB.

Este servidor es de uso gratuito y las bases de conocimiento alojadas allí son compartidas con toda la comunidad GeneXus, eliminándose automáticamente después de 15 días de inactividad.

Aquí ponemos nuestro usuario GeneXus y presionamos Send. Se creará en el Server una KB a partir de la nuestra, con el nombre que aquí indiquemos. Si ya existiera en el server una KB con ese nombre, fallará el envío. Debemos asegurarnos de que el nombre sea único.

KB Visualizer

The image displays two overlapping screenshots of the GeneXus Server web application. The background screenshot shows the login page with the title 'GeneXus Server' and the subtitle 'Team development for GeneXus'. It includes an 'Authentication Type' dropdown set to 'GeneXus Access', a 'Username' field containing 'diegomm', a 'Password' field with masked characters, and a 'Remember me on this computer' checkbox. The foreground screenshot shows the 'My Activity' dashboard. The top navigation bar includes the user name 'diegomm' and the language 'English'. The dashboard features a search bar and a table of activities. The table has columns for 'Knowledge Base', 'Operation', 'Date', and 'Comment'. A single activity is listed for the 'TravelAgency' knowledge base, with the operation 'Publish' performed on '07/22/20 04:44 PM', resulting in the comment 'Send KB'.

Knowledge Base	Operation	Date	Comment
TravelAgency	Publish	07/22/20 04:44 PM	Send KB

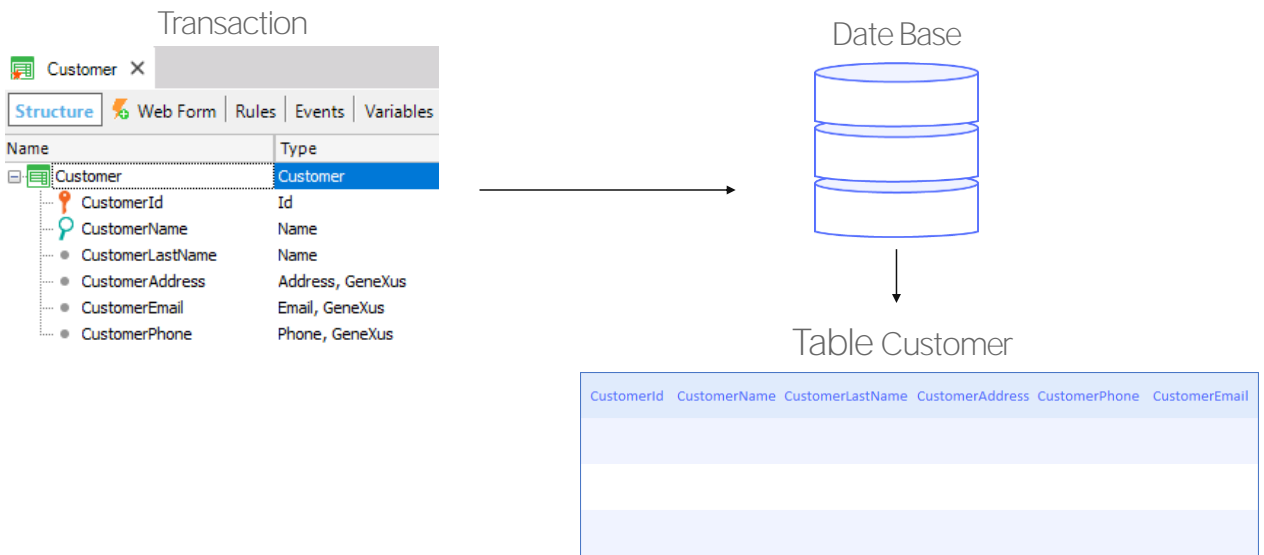
Desde nuestro navegador, podemos acceder a un visualizador de las KB en el servidor mediante la dirección que se nos proporcionó para enviar nuestra base de conocimiento.

Accedemos mediante nuestro usuario y contraseña de GeneXus. Para visualizar la KB que acabamos de subir, abrimos el menú "My account" y seleccionamos "Activity"

Allí vemos nuestra base de conocimiento. Podemos visualizar el tipo de operación que hicimos sobre ella, la fecha que realizamos la acción y los comentarios realizados en caso que le hayamos asignado alguno.

En resumen, hemos visto:

Al analizar las transacciones definidas, GeneXus determina las tablas físicas a crear.



Resumiendo lo que vimos: a partir del objeto transacción GeneXus construye un programa para crear la tabla que va a almacenar en la base de datos la información que manejará esa transacción.

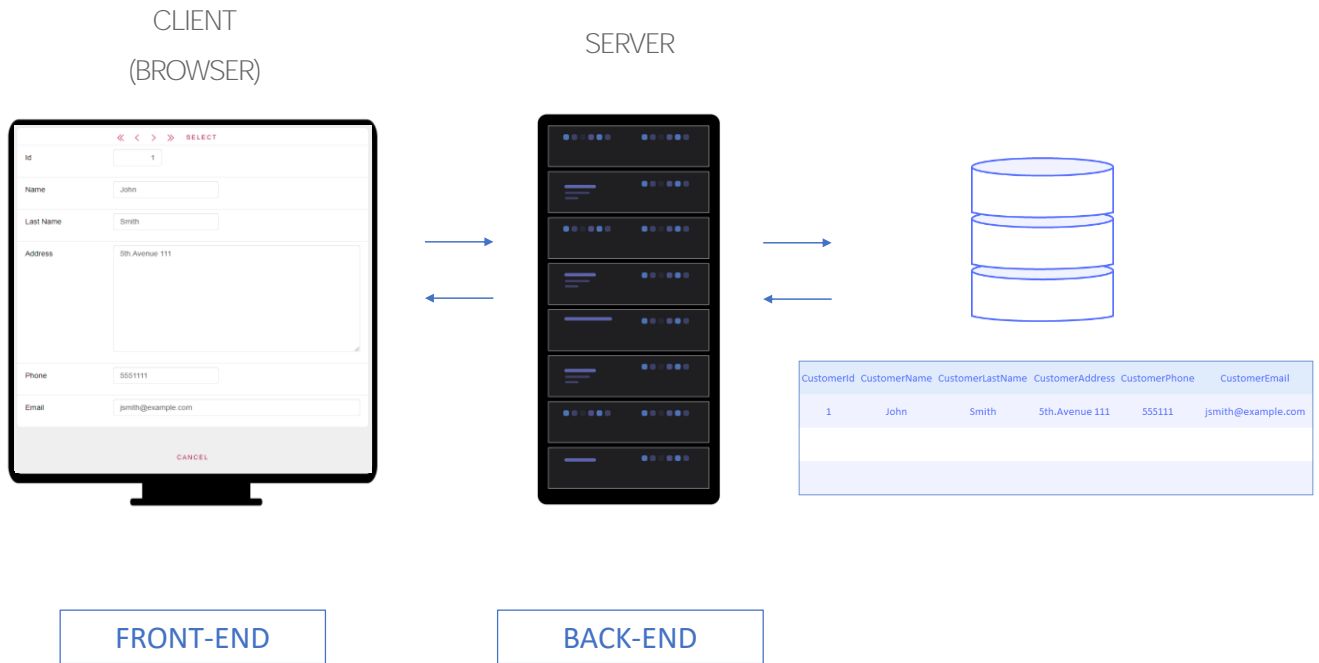
Cuando presionamos Create, es cuando ese programa se ejecuta.

Como en nuestro caso prototipamos en la nube, entonces nuestro servidor web y nuestro manejador de base de datos no estarán locales en nuestra máquina, sino en esa nube. Por tanto, cuando presionamos Create el programa de creación se enviará a la nube y se ejecutará allí, para tener por resultado esa tabla creada en base de datos en la nube.

Pero GeneXus, además de eso, envió al server el programa que construyó (en la plataforma elegida) para ejecutar la transacción. Ese programa, cuando se ejecute, tendrá una parte corriendo en el Server y otra parte corriendo en el Browser.

Cuando abrimos la transacción, vemos la interfaz de usuario, que nos permite realizar acciones como ingresar datos de un cliente nuevo.

La interfaz que estamos viendo en ejecución corresponde a esta parte de la transacción GeneXus. Podríamos personalizar este form. Lo veremos luego.



Al ingresar datos de un nuevo cliente y presionar Confirm la parte del programa ejecutando en el Browser se comunica con la parte del programa ejecutando en el Server, a quien le envía los datos, y el programa en el Server se comunica con la base de datos, pidiéndole que ingrese ese nuevo cliente en la tabla Customer que se había creado antes.

Más específicamente, cuando el usuario ingresa un valor en el identificador, la parte de la transacción ejecutando en el Browser se comunica con la parte de la transacción ejecutando en el Servidor, quién a su vez se comunica con la base de datos para saber si existe ese cliente, o es nuevo. Si existe, entonces la base de datos le envía los datos del cliente al Servidor, que se lo envía al Browser, quien los muestra en el formulario.

Por tanto, tenemos una parte del objeto transacción que será programado en el backend (y, por tanto, ejecutará en el server) , y una parte del objeto que compondrá el frontend (y por tanto, ejecutará en el cliente).

GeneXus[™]

training.genexus.com
wiki.genexus.com