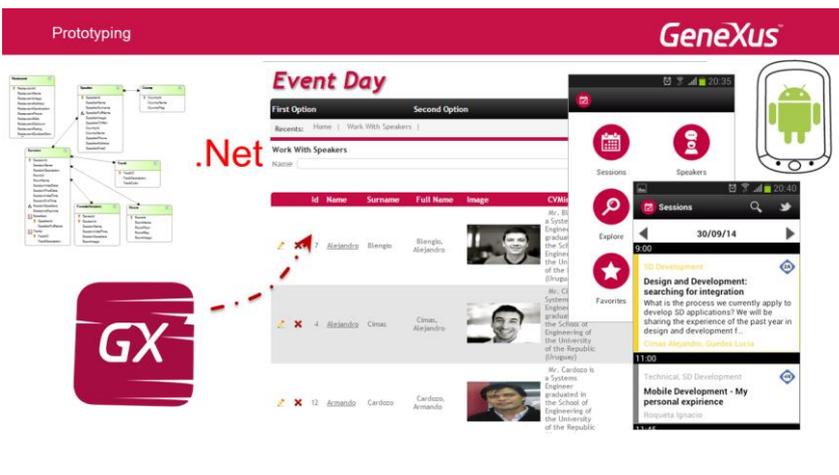


Prototipado y ejecución de aplicaciones SD con GeneXus



En nuestro proceso de construcción de una aplicación para manejar un evento a realizarse en ciertos días y lugar, hemos creado una KB en GeneXus con las Transacciones necesarias, y a partir de ella la aplicación Web y la aplicación para Smart Devices, para ser utilizadas por el personal que organiza el evento.



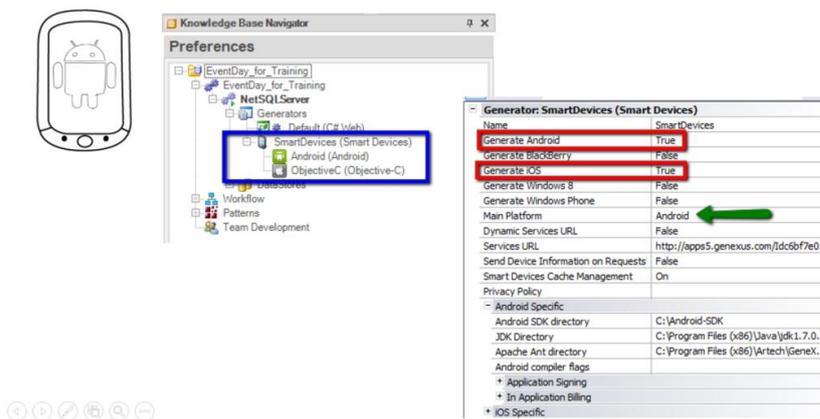
Hasta ahora hemos modelado, desarrollado y prototipado aplicaciones, generando en .Net para la aplicación web y en Android para la aplicación mobile.

Pero GeneXus nos provee otros mecanismos para prototipar fácilmente una aplicación para SD, ya sea usando otros emuladores o ejecutando directamente en el dispositivo, dependiendo esto también de la plataforma mobile elegida.

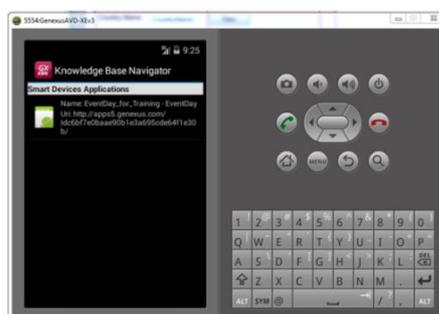
Veamos en el primer caso, la generación para dispositivos Android.



Cuando se aplicó el pattern Work With Smart Devices a una transacción, se agregó automáticamente el generador para Smart Devices. Si vamos a sus propiedades, vemos que por defecto se genera para Android y para IOS, y que la plataforma principal es Android.

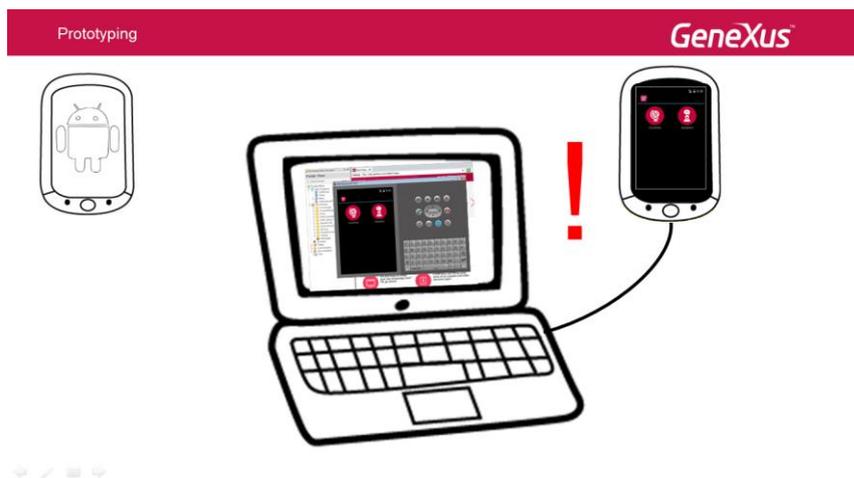


Por defecto, al ejecutar GeneXus abre un emulador de Android, para poder probar la aplicación, si es que no se tiene un dispositivo para hacerlo.

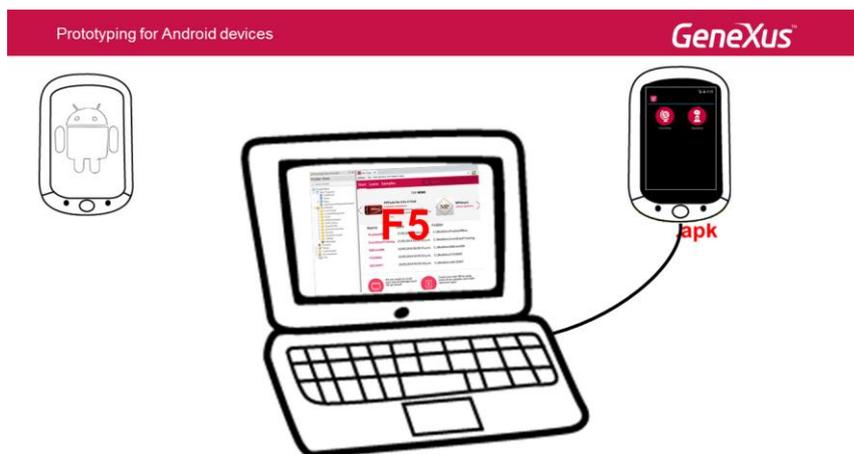


La mejor forma de prototipar es en el dispositivo. En el emulador no se puede probar todo, ya que ningún emulador se comporta exactamente igual que el dispositivo real. Un ejemplo de esto es si, queremos usar la cámara, grabar un video, grabar audio, etc.

Si se tiene un dispositivo con Android conectado al equipo de desarrollo mediante un cable USB, GeneXus detecta que el dispositivo está conectado y no abre el emulador. Si se tiene el dispositivo conectado al computador e igual se abre el emulador, puede ser problema de drivers, o por no estar habilitada la opción Depuración de USB, en el ítem Desarrollo de Configuración/Aplicaciones del dispositivo. Por favor diríjase al wiki de GeneXus por ese tema.



Con el dispositivo Android conectado, si se indicó un objeto de Startup, GeneXus genera la aplicación, la compila y la instala en el dispositivo. Cada vez que se presiona F5 en GeneXus, se actualizará la aplicación en el dispositivo, a través del cable.

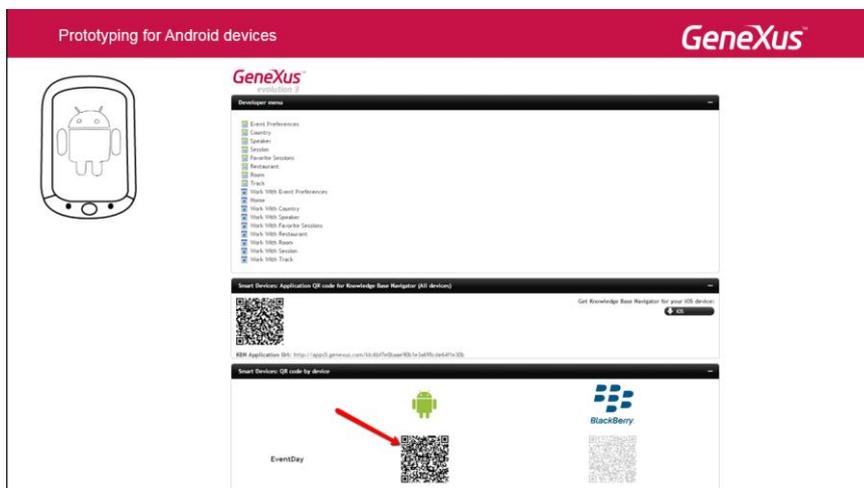


En caso contrario de que no se haya definido un objeto SD como startup, GeneXus instala el software Knowledge Base Navigator en el dispositivo mediante el cable y ejecuta la aplicación interpretada, a través del KBN.



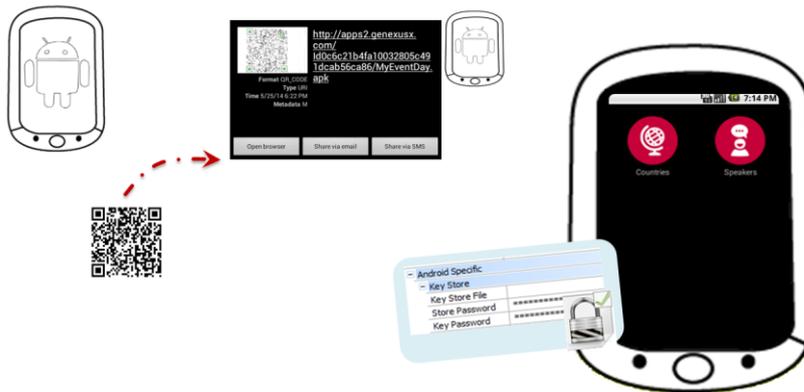
Cada vez que se presiona F5, se actualiza el contenido de la aplicación y se ven reflejados los cambios en la ejecución.

De las 2 opciones, es mejor ejecutar la aplicación compilada, lo que nos permitirá probar todas las funcionalidades en forma más realista, inclusive la prototipación de aplicaciones offline, que con el KBN no es posible. Debido a ésta y otras razones, el KBN de Android se está usando cada vez menos y se prefiere probar con la aplicación compilada.



Recordemos que otra posibilidad para instalar en el dispositivo la aplicación compilada, es leyendo con el dispositivo el código QR del paquete compilado, en la pantalla del Developer Menu.

Este instalará en el dispositivo un archivo de extensión apk, que contiene toda la lógica de la aplicación.



Pero para poder instalar aplicaciones en los dispositivos, las mismas deben estar **firmadas**. En el caso de Android, el apk siempre se genera firmado, no se necesita hacer nada adicional para instalar la aplicación en el dispositivo.

Veremos ahora cómo prototipar en otras plataformas, comenzando con IOS.



Para prototipar en dispositivos como el iPhone o el iPad que utilizan sistema operativo IOS, primero asignamos los valores a las propiedades del generador de Smart Device.



Generator: SmartDevices (Smart Devices)	
Name	SmartDevices
Generate Android	True
Generate BlackBerry	False
Generate iOS	True
Generate Windows 8	False
Generate Windows Phone	False
Main Platform	iOS
Dynamic Services URL	False
Services URL	http://apps5.genexus.com/Idc6bf7e0baae90b1e3a695cde64f1e30b/
Send Device Information on Requests	False
Smart Devices Cache Management	On
Privacy Policy	
- Android Specific	
Android SDK directory	C:\Android-SDK
JDK Directory	C:\Program Files (x86)\Java\jdk1.7.0_65
Apache Ant directory	C:\Program Files (x86)\Artech\GeneXus\GeneXusXE\3U2\Tools\ApacheAnt
Android compiler flags	
+ Application Signing	
+ In Application Billing	
- iOS Specific	
Execution Type	Knowledge Base Navigator (Device)
Execution Device	iPad Rodolfo



En este caso además se cuenta con diferentes opciones para ejecutar la aplicación, por eso es necesario indicar la forma de ejecución, configurando la propiedad **Execution Type** dentro del grupo de propiedades **iOS Specific**.

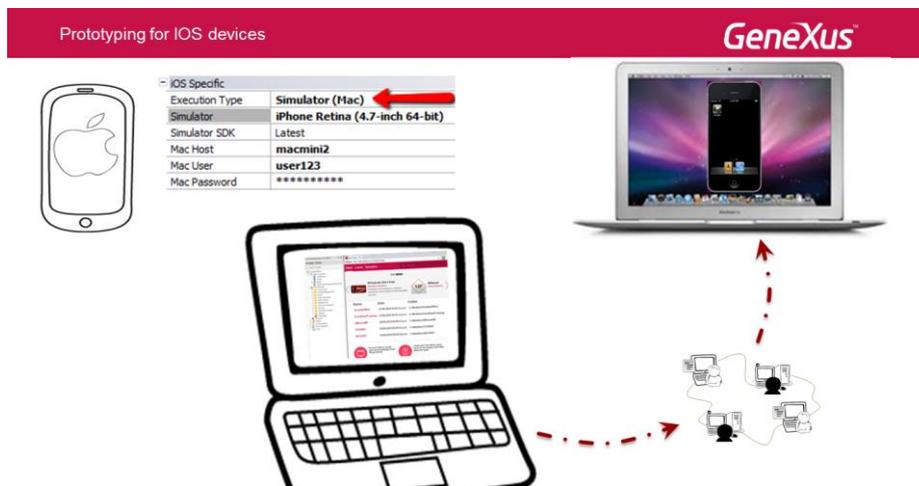
iOS Specific	
Execution Type	Knowledge Base Navigator (Device)
Execution Device	Simulator (Mac) iOS Device (Mac) iTunes Sync (Local) Build IPA (Local) Build for Distribution (Local) Knowledge Base Navigator (Device)

La opción **Simulator (Mac)** permite que la aplicación se ejecute en un emulador del dispositivo elegido, dentro de un computador Mac.

iOS Specific	
Execution Type	Simulator (Mac)
Simulator	iPhone Retina (4.7-inch 64-bit)
Simulator SDK	Latest
Mac Host	macmini2
Mac User	user123
Mac Password	*****

Se deberá configurar el dispositivo que se desea emular, el nombre del computador Mac donde se va a enviar la aplicación y las credenciales para conectarse, o sea el usuario y la contraseña.

En la Mac se deben instalar los componentes necesarios para poder ejecutar este tipo de aplicaciones y debe estar conectada a la misma red que el equipo donde se está desarrollando la aplicación en GeneXus.



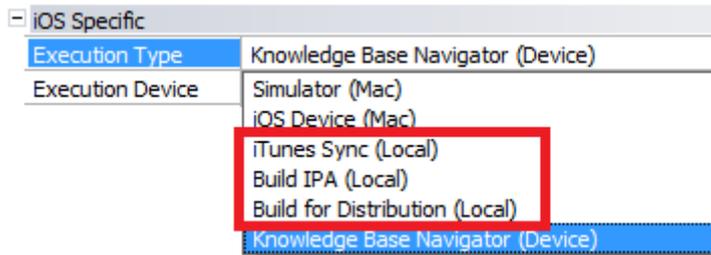
Al dar F5 se genera el proyecto Xcode, se envía a la MAC y se ejecuta en el emulador dentro de la MAC. Aquí no se firma la aplicación, ya que la misma se ejecuta en el emulador.

La opción **IOS Device (Mac)** permite que en caso de tener un dispositivo conectado a la Mac, se envíe el proyecto Xcode a la Mac, se compile, se genere el paquete de extensión .ipa y se instale en el dispositivo, para poder ejecutar la aplicación en el mismo.



En este caso se debe tener la firma instalada en la MAC y el dispositivo debe estar debidamente autorizado.

Las opciones marcadas como (local), Itunes Sync, Build IPA y Build for Distribution, permiten transferir el proyecto Xcode a la Mac, compilarlo y generar el paquete ipa, para luego copiarlo al equipo de desarrollo. La ventana de Output del IDE de GeneXus muestra donde quedará copiado el archivo.



En el caso de **Itunes Sync**, luego del que el ipa es copiado al equipo de desarrollo, se abre automáticamente el iTunes para mostrar la aplicación en su catálogo, lo que permite poder instalarla en el dispositivo conectado al PC mediante un cable.



En el caso de **Build IPA**, se copia la aplicación compilada a la carpeta del modelo GeneXus, pero no se abre el iTunes.



Y en **Build for Distribution**, se compila la aplicación en la Mac utilizando la firma de distribución y se copia el ipa generado, a la carpeta del modelo de GeneXus.



Cuando seleccionamos **Knowledge Base Navigator (Device)**, la aplicación se genera y se ejecuta en el dispositivo a través del KBN, el cual deberá descargarse del Apple Store e instalarse en el dispositivo. Esta opción permite ejecutar una aplicación para IOS cuando no tenemos una Mac, ya que la misma no se compila y se ejecuta en forma interpretada a través del KBN.



Con este procedimiento, la prototipación es inalámbrica. El dispositivo debe estar conectado a la misma red wireless que el equipo de desarrollo y debe tener habilitadas las notificaciones. Luego de que el usuario se loguea en el dispositivo con su usuario gtechnical, el dispositivo queda registrado como asociado al perfil de usuario y lo podremos elegir para completar la propiedad Execution Device. Cada vez que presionamos F5 en GeneXus, la aplicación se actualizará automáticamente en el dispositivo.

Ahora veremos cómo prototipar en Windows Phone y Windows 8.



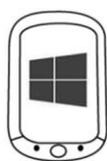
Trataremos juntos a los generadores Windows 8 y Windows Phone, debido a su similitud.

Si queremos generar una aplicación para Window 8 o Windows Phone, para empezar seteamos la propiedades correspondientes del generador Smart Devices, es decir la propiedad Generate Windows 8 (o Windows Phone) en True y la propiedad Main Platform en el valor Windows 8 (o Windows Phone) respectivamente.



Generator: SmartDevices (Smart Devices)	
Name	SmartDevices
Generate Android	True
Generate BlackBerry	False
Generate iOS	True
Generate Windows 8	True
Generate Windows Phone	False
Main Platform	Windows 8
Dynamic Services URL	False
Services URL	http://apps5.genexus.com/Idc6bf7e0baae90b1e3a695cde64f1e30b/
Send Device Information on	False
Smart Devices Cache Manag	On
Privacy Policy	
Android Specific	
iOS Specific	

En el caso de que la plataforma sea Windows 8 o Windows Phone, no hay propiedades específicas a setear.



La prototipación es similar a la de Android, si GeneXus detecta a un dispositivo con Windows 8 o Windows Phone conectado, copiará la aplicación al dispositivo y la ejecutará en el mismo. En caso de que no se detecte un dispositivo conectado se abrirá un emulador.

En el caso de estos generadores, siempre debe definirse un startup object, ya que la aplicación en el dispositivo se ejecuta únicamente en forma compilada, por lo que no se puede utilizar el KBN.

Tanto se utilice el emulador como el dispositivo, el equipo de desarrollo tendrá que tener instalado el SDK de Windows Phone o de Windows 8, según el caso.

Además deberá tener instalado el Visual Studio para ambos casos, aunque alcanza con la versión gratuita (Visual Studio Express). Por más información sobre los requerimientos, consulte en los sitios que se muestran en pantalla.

<http://wiki.genexus.com/commwiki/servlet/hwikibypageid?26848>

<http://wiki.genexus.com/commwiki/servlet/hwikibypageid?20715>

Por último, veremos la protipación para la plataforma Blackberry.



La forma de prototipar con Black Berry es básicamente la misma que con Android y con plataformas Windows. Para empezar seteamos la propiedades correspondientes del generador Smart Devices, es decir la propiedad Generate Blackberry en True y la propiedad Main Platform en el valor BlackBerry.



Generator: SmartDevices (Smart Devices)	
Name	SmartDevices
Generate Android	True
Generate BlackBerry	True
Generate iOS	True
Generate Windows 8	False
Generate Windows Phone	False
Main Platform	BlackBerry
Dynamic Services URL	False
Services URL	http://apps5.geneXus.com/1dc6b7e0baae90b1e3e695cde64f1e30b/
Send Device Information on Requests	False
Smart Devices Cache Management	On
Privacy Policy	
- Android Specific	
Android SDK directory	C:\Android-SDK
JDK Directory	C:\Program Files (x86)\Java\jdk1.7.0_65
Apache Ant directory	C:\Program Files (x86)\Artech\GeneXus\GeneXusEv3U2\Tools\ApacheAnt
Android compiler flags	
+ Application Signing	
+ In Application Billing	
- BlackBerry Specific	
BlackBerry SDK directory	C:\Program Files\Research in Motion\BlackBerry JDE 4.6.1
BlackBerry Simulator	9700
BlackBerry Simulator Directory	C:\Program Files\Research in Motion\BlackBerry JDE 4.6.1\simulator
BlackBerry Signature Tool Password	*****
BlackBerry BIS-B Connection String	
BlackBerry OS Version	5.0.0
- iOS Specific	

Al igual que en Android, si al ejecutar la aplicación no se detecta un dispositivo conectado, se abre un emulador de BlackBerry.



Si se conecta un dispositivo BlackBerry al computador, GeneXus detecta al dispositivo conectado y si no se definió un objeto de startup se instala el KBN y se ejecuta la aplicación en forma interpretada.

En caso contrario, se transfiere la aplicación compilada, que es un archivo con extensión .jad.

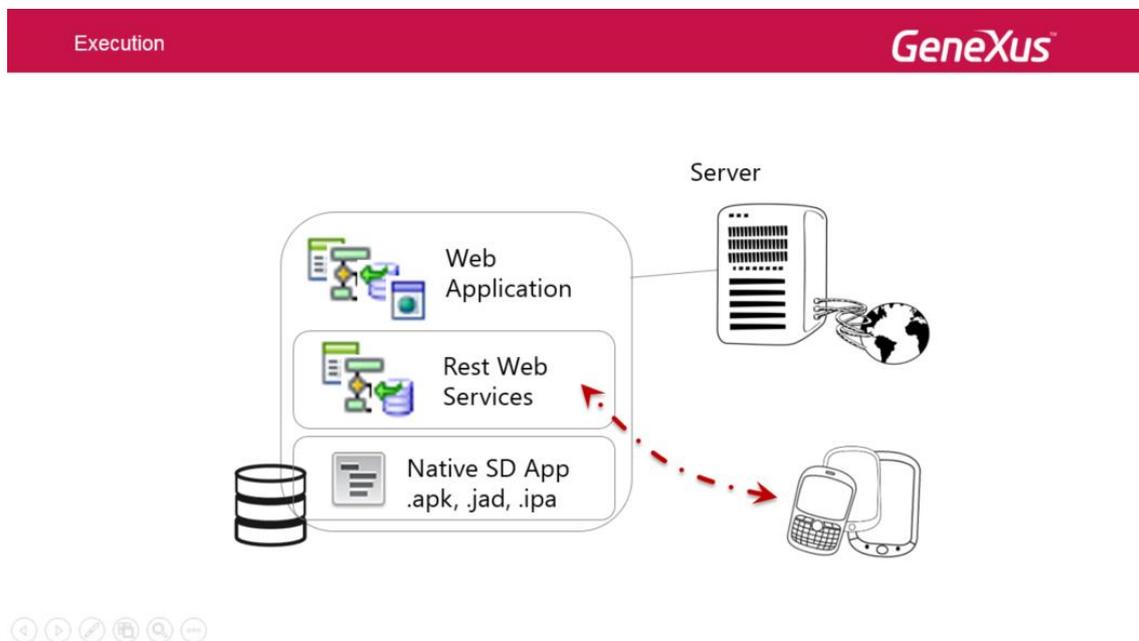
También es posible instalar la aplicación compilada desde el Developer Menú, utilizando el QRCode correspondiente.

Cuando se genera una aplicación para BlackBerry, la misma no está firmada, pero el proceso de pedir una firma es sencillo y gratis.

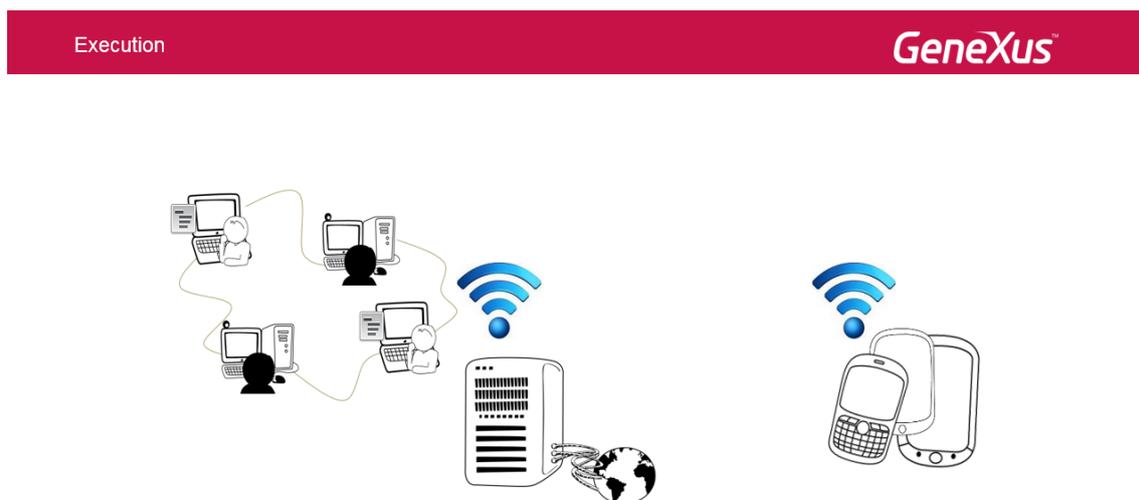


Por más información, lo invitamos a visitar la dirección que se muestra en pantalla.

Una vez instalada la aplicación en el dispositivo, veamos las diferentes opciones que tenemos para **ejecutarla**, o sea, como poder acceder a los Servicios Rest desde la aplicación, instalada en el dispositivo.



Una posibilidad es tener los Servicios Rest instalados en un equipo dentro de una Red Privada con acceso Wifi o en un servidor disponible vía Internet. De esta forma la aplicación para Smart Devices instalada en el dispositivo, se conectará a esa Red Privada o a la Web via Wifi para acceder a los servicios Rest.



Otra alternativa, es configurar una red Computer to Computer, también conocidas como red ad hoc. Esta posibilidad del sistema operativo Windows, nos permite establecer una conexión WiFi, directamente entre dos computadoras o una computadora y un dispositivo con conexión WiFi. En este caso nuestro equipo de desarrollo servirá como web server.

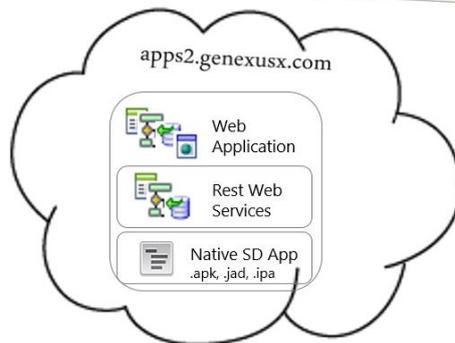


Para más información sobre cómo configurar este tipo de redes, puede visitar el link en pantalla: <http://windows.microsoft.com/es-419/windows/set-computer-to-computer-adhoc-network#1TC=windows-7>

Por último, tenemos una opción en GeneXus para ejecutar una aplicación, instalándola en la Nube, con solo un clic.

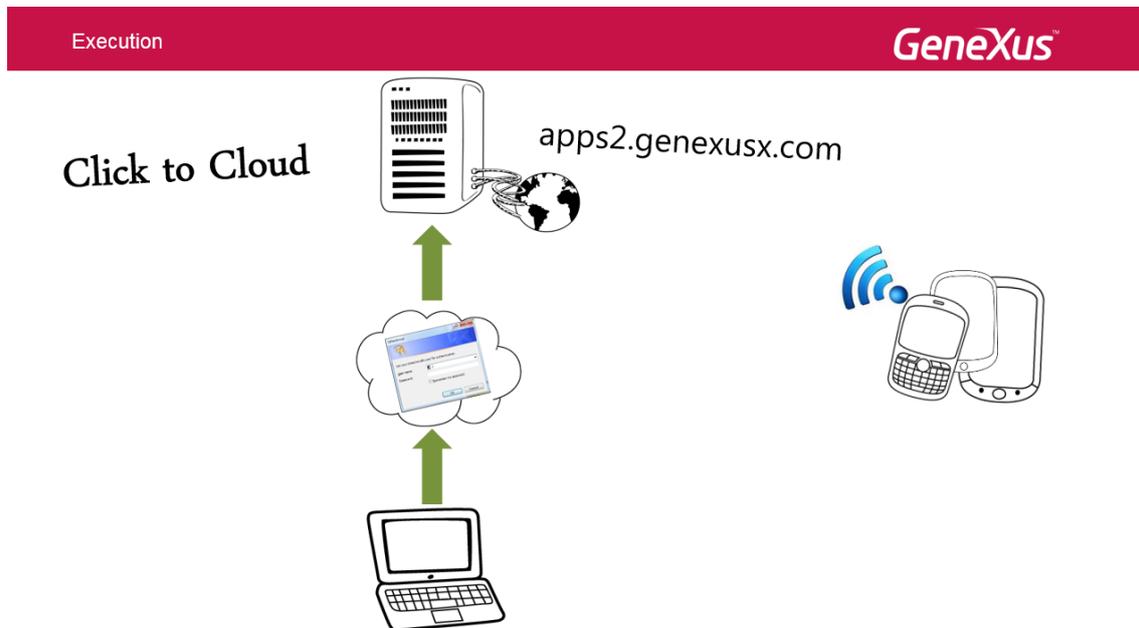
Deploy to Cloud

Execution	
Deploy to cloud	Yes
Deploy Server URL	http://apps2.genexus.com
Deploy Virtual Directory	Id985e526c2e81461796b3da90b59c9c5a



Esto se logra simplemente configurando la propiedad Deploy To Cloud del generador Default con el valor Yes. Al dar F5, la aplicación Web se instala en la Nube, además de instalarse la aplicación para Smart Devices en el dispositivo.

Por defecto se utiliza el servidor apps2.genexusx.com en la nube de Amazon, disponible especialmente para prototipar aplicaciones GeneXus.



Al generar la aplicación, se solicitarán las credenciales de GXtechnical para poder instalarla en la Nube.

Una vez instalada, ya es posible accederla desde el dispositivo consumiendo los Servicios Rest directamente desde la aplicación en la nube.

En este video vimos como prototipar y ejecutar nuestra aplicación para Smart Devices.

Luego de que tengamos pronta la misma, nos interesa publicar la aplicación en un servidor propio o en los respectivos stores. Eso lo veremos en el próximo video.