

GeneXus™
The power of doing

GeneXus

Update to GeneXus 16

GeneXus Training Program

GeneXus™ 16

Ahora vamos a arrancar con un OverView, por lo tanto es simplemente un paneo por algunas de las cosas principales de GeneXus 16 y después ustedes sí van a poder profundizar en cada uno de los temas, en la mayoría de los temas tiene sentido profundizar y ahí profundizarán, otros quedarán simplemente por aquí ...

Overview

Como saben, los pilares principales de GeneXus 16 vienen por el tema de multi experiencias, que vemos que las aplicaciones no van a ser solamente mobile y web como las vemos hoy, sino que van a estar disponibles en más dispositivos, para eso está preparada esta versión. Y lo otro, es que también vamos a estar incorporando componentes con inteligencia artificial que van a ayudar a optimizar procesos a través de las soluciones o también ayudar y apoyar en la toma de decisiones.

Así que esos pueden ser componentes que nos ayuden, los podemos agregar a las soluciones ahora en GeneXus 16 y a partir de ahí brindar soluciones que dan más valor a nuestros usuarios.

Eso es en una línea lo que es GeneXus 16.

GeneXus
Platform

Solutions

... ahora, esas soluciones nosotros por un lado podemos dividir lo que son las soluciones en si y por otro lado en cómo las construimos.



Si la ordenamos así y vemos de alguna forma... todo esto es Dev/Ops, o sea, el cómo nosotros las construimos. Tenemos todas las herramientas para construirlas. Después, ésta es la solución que nosotros vamos a estar desarrollando, puede tener múltiples componentes como están viendo y eso de vuelta implica los cuidados que tenemos que tener en seguridad, las funcionalidades que tiene que tener GeneXus para la integración para que ustedes puedan integrar todo este tipo de soluciones. Y después, devuelta otra parte de operaciones digamos ,que tenemos que contemplar y que contemplar en GeneXus es que nos ayude a publicar esas soluciones. Así que: primero el desarrollo (el qué es lo que queremos enfocar) y después, solucionar y en cómo las desplegamos.

DevOps practices

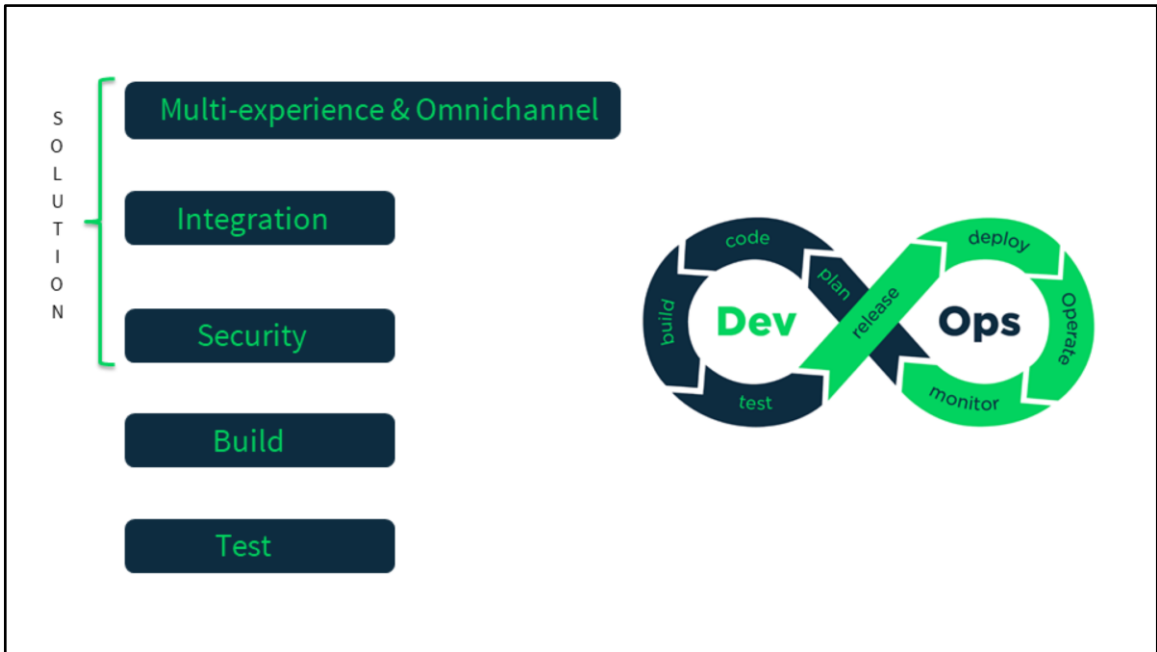


En todo esto, nosotros queremos seguir un ciclo DevOps o queremos facilitar esa cultura en las empresas.

Nosotros mismos utilizamos eso, empezamos a comenzar en transitar ese camino automatizando lo automatizable internamente en la construcción de GeneXus y muchas de las empresas de ustedes ya también están empezando a tomar esa cultura, hay muchos incentivos en varias empresas, muchas veces las más jóvenes que ya empiezan a trabajar así, pero ahora están las condiciones dadas para que todos podamos adoptar eso.

En este ciclo están todas estas etapas, y en éstas etapas van a ser mas o menos la guía por el cual también vamos a recorrer las funcionalidades de GeneXus

Development



Así que arrancamos con lo que es el desarrollo.

Vamos a estar construyendo una solución como yo les decía : Multi-experiencie & Omnichannel, vamos a tener que tomar en cuenta conceptos de Integración (Integration) y de seguridad (Security)

Multi-experience & Omnichannel

Integration

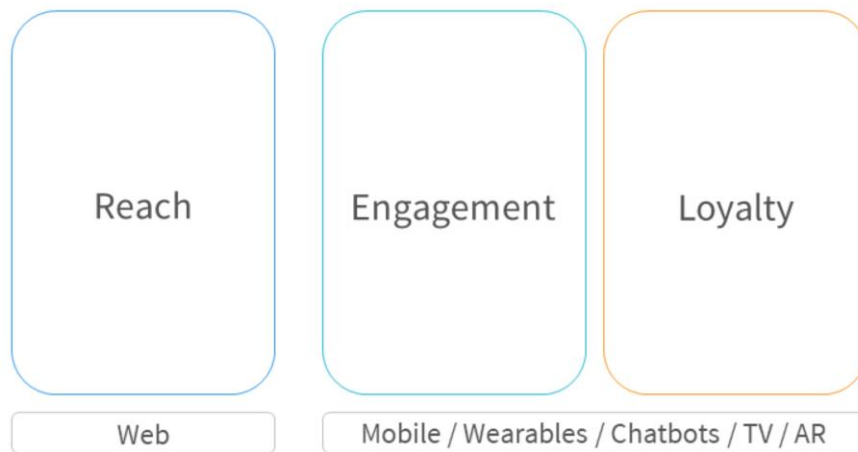
Security

Build

Test



...y después vamos a ver como las construimos y las testeamos.



En el primer punto, cada uno de esos canales tiene un fin específico, no es que el web les va a solucionar todo, no es que ahora tienen que tirar todo el web y va a ser solamente mobile o la tendencia que ustedes quieran escuchar, sino que, en realidad cada uno tiene un fin específico y lo bueno con GeneXus es que podemos crear cada uno de esos componentes para ese fin y entonces tener una solución que es mucho más abarcativa.

Multi-experience & Omnichannel



- Powering Design
- UX: Streams of information (Twitter, Facebook-like apps)
- Conversational UI: Chatbots
- Multi-layer Maps
- Powering analytics
- Woreables (Apple Watch, Apple TV)
- ... and more in Web and SD

Hemos estado dando más poder o más facilidad al diseño en GeneXus y distintas cosas, mejorando la experiencia de usuario, agregando chatbots también y cada uno de esos puntos que ahora vamos a ver.

Powering Design

Design Systems

Principles

Patterns

Practices

¿Para qué están hechos los Design Systems?

Están pensados para cuando yo tengo múltiples canales, ahí surgió mucho mas la importancia todavía. Yo tengo que lograr una coherencia y una dirección en mi diseño, no puede ser que cada uno tenga un diseño que no tenga nada que ver con el otro, entonces para lograr dirección y coherencia es que están los Design Systems y que son un conjunto de principios, prácticas y patrones para lograr esa coherencia.

Design Systems



Fiori Launchpad

Fiori Split Screen
Master List

Country (4)		Country
Search		Uruguay
China	1380996000	GENERAL
Uruguay	3440157	Id: 2 Name: Uruguay Population: 3440157
USA	325318000	
Iceland	334252	

Uno de esos Design Systems que ya están integrados en GeneXus y para el cual se hizo incluso una edición especial de GeneXus es el de "Fiori".

Ahí lo que están viendo es... (Fiori es el Design Systems de SAP) entonces, en el sitio de SAP, SAP explica cómo hay que seguir ese Design Systems, cuales son las guías que hay que seguir.

En GeneXus lo que fuimos agregando fueron: Patterns, temas y objetos específicamente para que se genere automáticamente siguiendo esas guías.

Acá lo que ven como un ejemplo es un Lunchpad de Fiori y esto es un Fiori Split Screen Master List, o sea, después hay diez tipos de vistas distintas y cada una de esas tiene su Patterns o su forma de construirse... pero estos son ejemplos con este nivel de detalle que los botones tienen que estar acá abajo, que aquí se tiene que poder navegar para atrás, digamos, esto así sigue esa guía del Design Systems.

Design system

Base styles



Themes

Controls



Stencils



Patterns

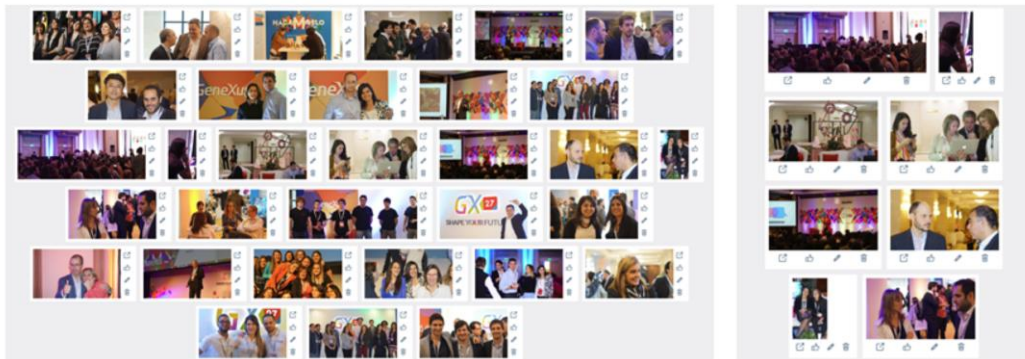
Master Pages

Entonces... en GeneXus tenemos nuevos conceptos para poder describir en una base de conocimientos ese Design Systems y generar código a partir de ahí.

Por estos tres temas vamos a entrar en detalle, después Federico Salomón va a entrar en detalle.

UX

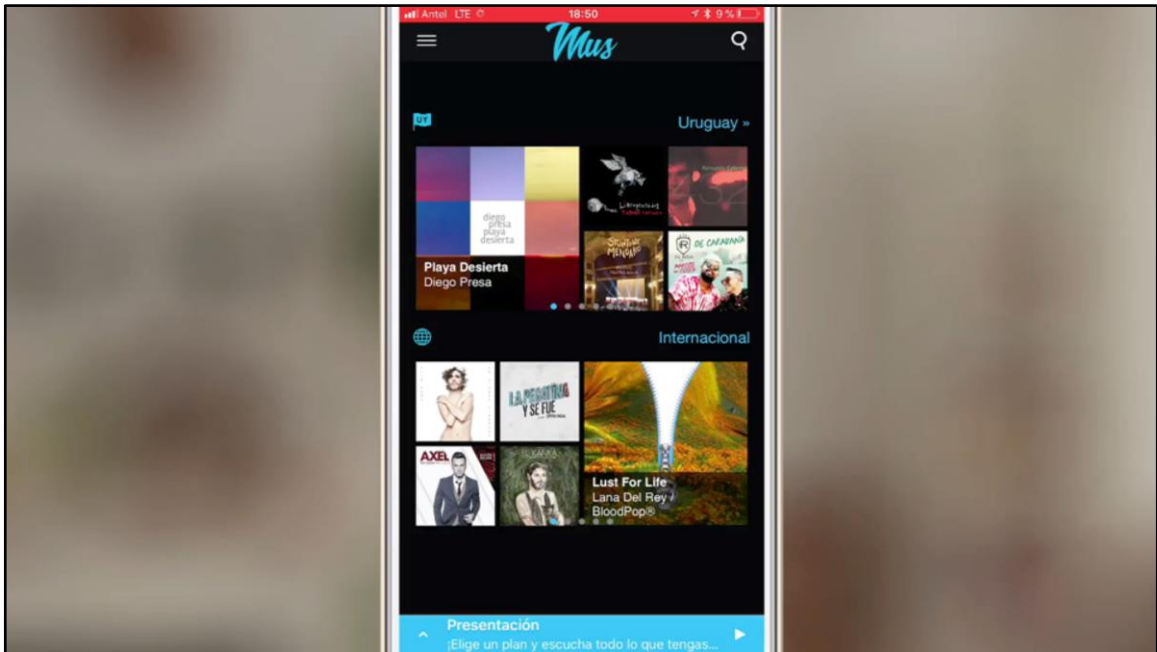
Y después, mejoras en la experiencia de usuario...



Hemos trabajado en web, esto lo que ven acá, se logra con un Flex grid.

Como ven, no es que esté fijo a 12 columnas, es mucho más flexible que solamente responsive web design.

Cada uno de estos componentes se puede anclar a ciertas partes de la pantalla y automáticamente si uno ensancharía la pantalla, entonces éste componente de acá empieza a verse acá arriba, empieza a reordenarse cada uno o si lo veo en una pantalla más angosta, entonces los van tirando para abajo a los componentes, eso está tanto para web como para Smart Devices.



Acá el video de algunas aplicaciones que se hicieron utilizando los nuevos conceptos que se fueron agregando a GeneXus, desde GeneXus 15 hasta GeneXus 16 con distintas funcionalidades que dan hasta audio streaming, video streaming, las transiciones las pueden gobernar a ustedes, así como en Power Point las pueden gobernar en GeneXus, lo mismo las animaciones, interacciones ahí en este caso con mapas (de eso también vamos a entrar en profundidad luego)

... esto que ven ahí era integración con el dispositivo, con todo lo que es Face recognition y controles de biometría para autenticación.

Ustedes pueden tener componentes en sus soluciones específicas para eso.

Esto de recién era un chat, acá tienen integración con realidad aumentada, la realidad aumentada en si no esta integrada en GeneXus 16 todavía pero ustedes pueden agregar controles externos que sí lo tengan.

Estos son animaciones, son Lotties, se llaman... este es el formato, es un Json a partir del cual ustedes pueden definir una animación y reproducirla.

Esto fue para china específicamente, dos mil millones de personas usan allá una de estas dos aplicaciones, así que cualquiera que haga una aplicación para allá tiene que seguramente integrarse con estos medios de pago.

Después, esta es una aplicación a la que se agregaron conceptos de gamification, o sea, distintas personas apostando entre si y jugando a ver quien acierta mejor los resultados del mundial.

Esto es la aplicación de Tienda Inglesa (esta en el store) y es de una aplicación de retail de Uruguay a través de la cual se pueden hacer compras.

Para esta aplicación también se agregaron muchas funcionalidades para los dinamismos que requería.

Todo esto, con live editing es que se puede lograr ese nivel de detalle, porque sino nos aburriríamos de estar seteando propiedades y probando el resultado de esas propiedades a

ver como se ven en realidad

La única forma de poderse tomar el tiempo para lograr ese nivel de detalle es con live editing.

Conversational UI

Y después, el segundo punto es: Conversation UI para hacer interfaces conversacionales (bueno yo les mostraba) Fuimos agregando la potencialidad de GeneXus de poder hacer chats que son la base para hacer un chatbot, ese tipo de inferfaz.

Chatbots

Natural Language

Voice

Text

Hybrid

Do tasks

Ahora lo que hay detrás de la interfaz en realidad es una conexión con inteligencia artificial (AI) específicamente con el área de lo que es tratamiento natural de l lenguaje. Desde esa forma entonces, nosotros generamos un modelo en GeneXus a partir de un objeto en el cual se... en un objeto GeneXus se explica el detalle al flujo de una conversación y apartir de ahí GeneXus genera un modelo también que va a un motor que resuelve el tratamiento natural del lenguaje.

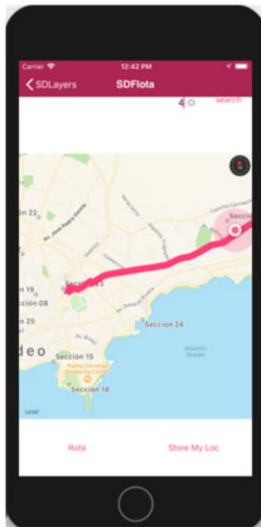
A través de las apis se puede interpretar la voz y ese texto entonces, se interpreta por ese motor.

Las aplicaciones GeneXus, además como tienen otros componentes... entonces en ese chat también no podemos solamente esperar que se haga una interacción a través de audio o a través de textos, sino que también el chatbot podría estar respondiendo pantallas... embebidas que son mucho más ricas, a veces explicar una gráfica es mucho más difícil que simplemente mostrarla en el chatbot, entonces ahí es otro valor de GeneXus, el de poder mostrar entonces componentes embebidos en ese chat.

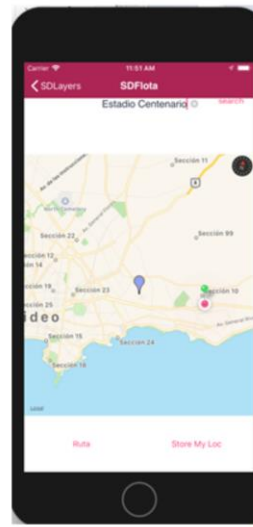
Y así como muestra un componente, también con una gráfica puedo mostrar un componente que tenga acciones donde por ejemplo, autorizo una compra o cualquier operación así, que pueda estar siendo necesaria, de eso vamos a entrar en detalle luego.

Multi-layer maps

Direction and Selection layer - SDmaps



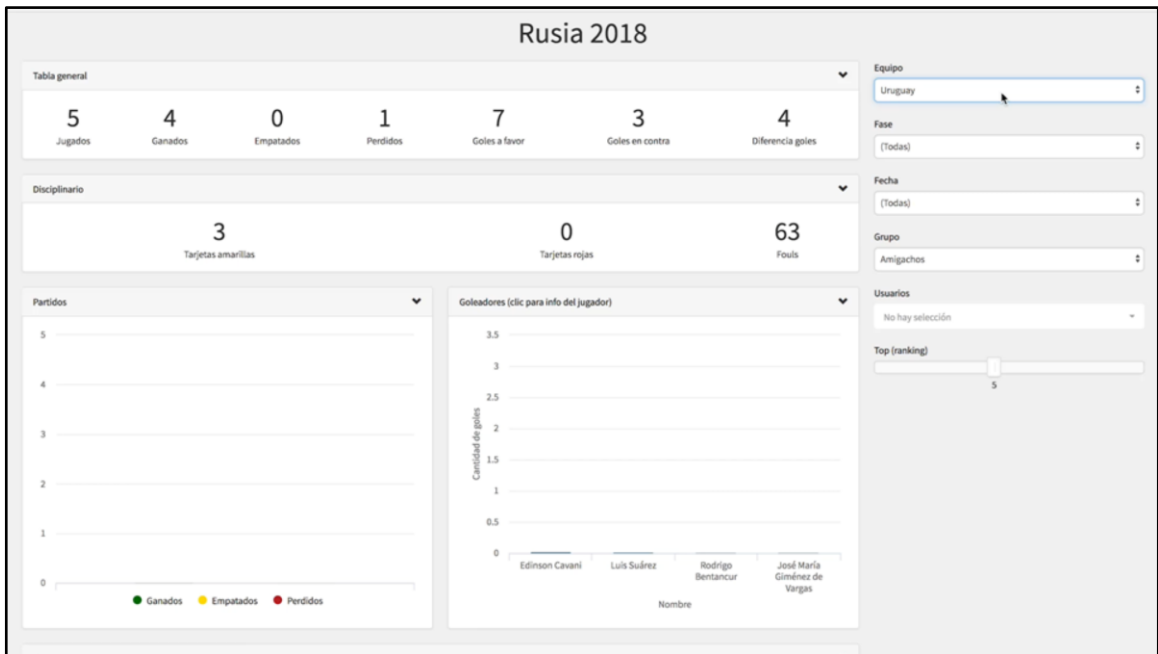
Direction
Layer



Selection
Layer

A nivel de mapas se pueden ahora dibujar capas arriba de los mapas, y vamos a estar agregando mas capas, pero las primeras que ya están disponibles es poder dibujar una línea, seguir una línea en un mapa o también usar un selector y después mover todo el mapa hasta seleccionar el punto, mover el punto hacia un lugar y seleccionarlo.
Javier Larrosa va a explicar más después de esto.

Powering analytics



Y después hemos trabajado en dar mas facilidad al análisis de los datos con un objeto dashboard en GeneXus.

Eso es lo que ven en GeneXus, esto lo que ven en pantalla es un objeto dashboard completo. En ese objeto dashboard se tiene un editor web y se fueron arrastrando query viewers y data providers que muestran esos resultados de una forma.

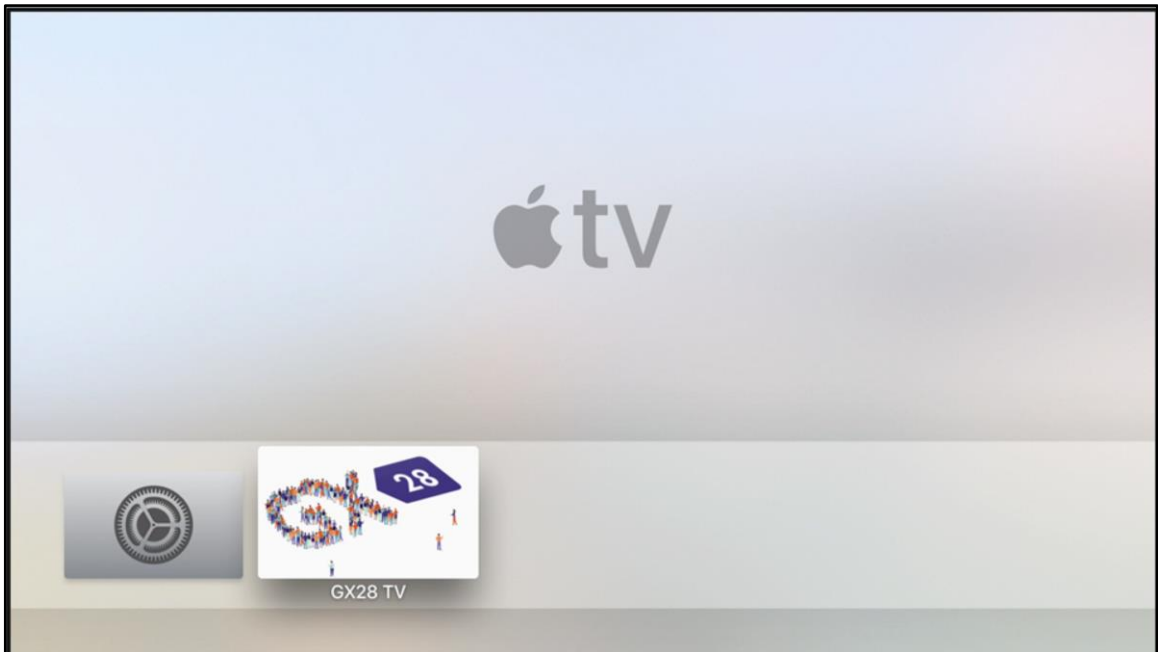
Un objeto dashboard es un business dashboard que sirve para mostrar Key performance Indicators (KPIs) entonces, cada una de nuestras soluciones puede tener un dashboard así y mostrar esos KPIs.

Apple Watch / Apple TV

Lo que agrega al tema del omnichannel son dos generadores, en realidad viene incluido con el generador de IOS, a esta altura ya mal llamado IOS porque además de eso ya genera para Apple Watch y Apple TV que son dos sistemas operativos distintos, o sea, IOS es para el Iphone o el Ipad, pero, después tvOS es para el Apple TV y watchOS para el Apple Watch. entonces, en un generador en realidad generamos para tres lenguajes tres sistemas operativos, pero en realidad se gobiernan con el mismo lenguaje que es Swift.



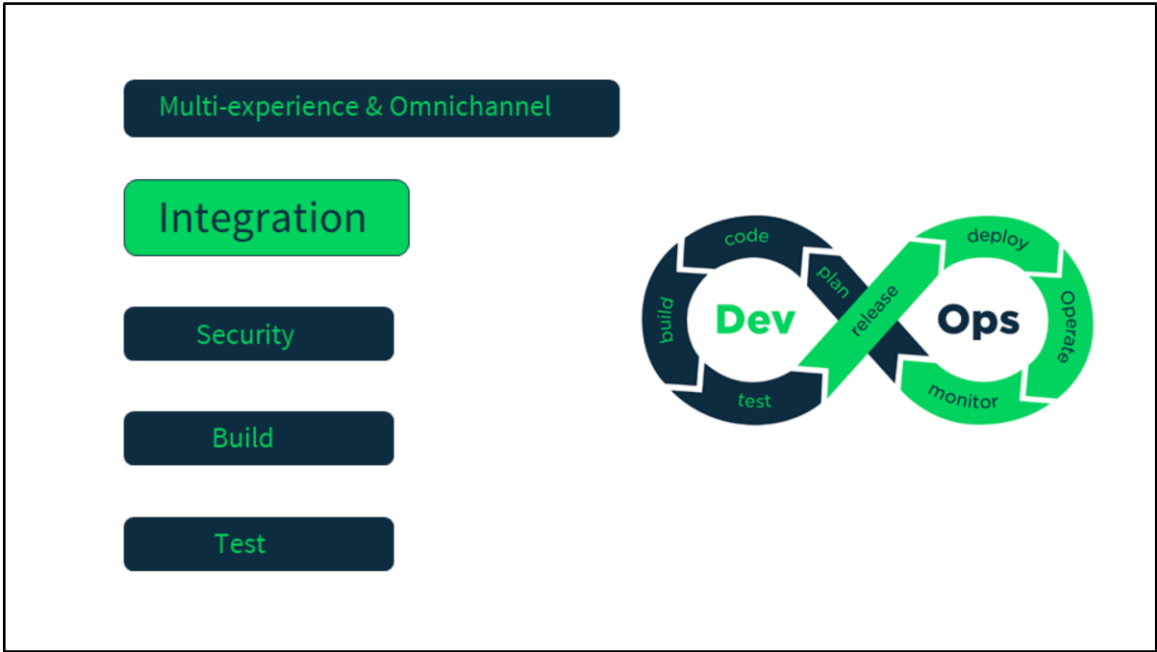
Esto es la aplicación del evento vista en el Apple Watch, lamentablemente solo hemos llegado a la etapa beta de la aplicación, no la hemos podido liberar para estar disponible durante el evento, pero ahí es como se veía, llegan notificaciones y se pueden ver las sesiones dentro del Apple Watch, las sesiones que están por venir dentro del Apple Watch. Esa es la funcionalidad que teníamos ahí.



Esto es como se vería la aplicación, esto es devuelta, una versión beta de la aplicación para Apple TV.

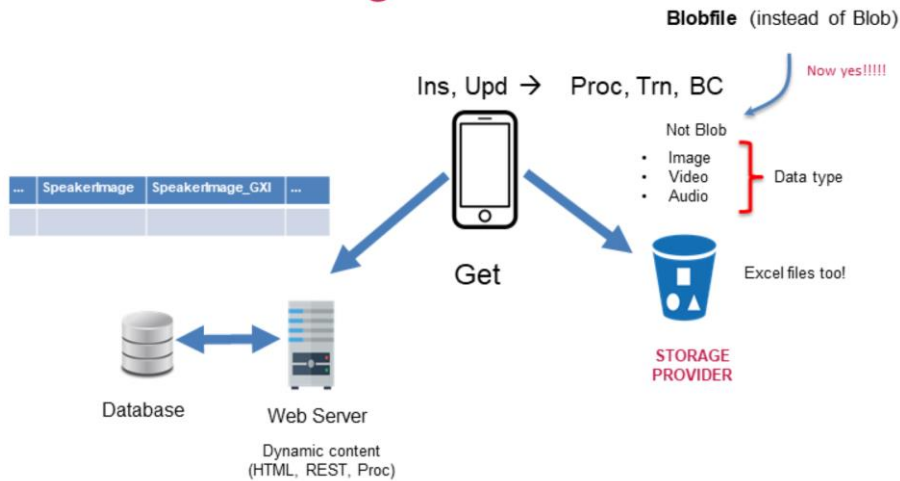
Como ven, uno navega entre los datos, con el cursor van para adelante, para atrás y después selecciona un valor, selecciona un campo, un componente y siempre es como que uno le va a dar click a botones para después ahí estar viendo el video.

Esta base de conocimiento se va a poner pública, estamos trabajando en eso para que ustedes la vean también como un ejemplo y de como se hizo toda la aplicación del evento, hay muchos componentes de interfaz de usuario que pueden ser entonces, analizados por ustedes para ver y replicarlo en sus soluciones, hemos puesto énfasis en eso.



Bueno, muy bien...
En segundo punto con respecto a integración.

Cloud external storage



La novedad ahora es, esto es external storage que venia con GeneXus 15, que ustedes pueden subir las imágenes, por ejemplo a un storage provider como Amazon S3 o en cualquiera de las plataformas, Bluemix o lo que sea. Lo que podían hasta ahora guardar acá eran imágenes, videos y audios.

Y ahora en GeneXus 16, algo nuevo que pueden hacer es... hay un nuevo tipo de dato BlobFile, el Blob lo que pasa que no tiene toda la información como para que nosotros pongamos en el campo una URL apuntando al external storage, entonces agregamos el tipo de BlobFile y ese si permite, o grabar localmente en la base de datos o grabar externamente en un storage provider.

Así que la recomendación en realidad es, a partir de ahora todo lo nuevo que hagan, y lo que puedan convertir por supuesto también, empiecen a usar BlobFile en vez de Blob, porque tiene mas información y les va a permitir en cualquier momento pasar a usar storage provider y de esa forma lograr una mayor escalabilidad de la solución.

Los archivos Excel también se generan en el storage provider, empezamos a generar archivos directamente en el storage provider de forma de usar lo menos posible el file systems local de la aplicación web, donde corre la aplicación web. Esto es también por razones de escalabilidad, si queremos estar ejecutando en contenedores, tenemos que saber que esos contenedores tienen que ser descartables, o sea, hay una base, hay una imagen base a partir de la cual se crea un contenedor, pero luego si cuando eso escala puede ser que ese contendor se destruya y se creen cinco contenedores mas, entonces, todo lo que se graba localmente tiene que ser descartable, es mas, lo tenemos que evitar grabar archivos localmente, tenemos que tratar de grabar todo externamente, la base de datos ya la tenemos afuera siempre, eso es normal, pero también tratemos de manejar los archivos afuera.

-¿Cuándo dijiste "Excel" significa cualquier tipo de documento? Por ejemplo, contratos que se quieran almacenar dentro del sistema en el storage...

- emm... en forma predeterminada no, pero hay funciones en el tipo de dato files y el tipo de datos external storage, con esos dos ahora puedes lograr que cualquiera de esos archivos se grabe externamente. El tipo de datos external storage tiene funciones de Upload File y de Download File para poder subir ahí, con esos archivos entiendo que todavía se van a grabar archivos intermedios una vez en el local file systems, pero ya con estas funciones enseguida lo pueden subir al S3 o a donde sea.
- O sea, que se graba en el sitio web y vos después con otro proceso tenés que mandarlo al storage?
- Sí, o en la misma función cuando lo grabas después ya usas el tipo de data ofile y el external storage, esos dos combinados y lo subís al S3 y después ya te da el link y con ese link después seguís procesando.
- ¿ No controla el tipo de datos eso?
- No, no! Ahí vos subís un binario cualquiera, lo que sea, incluso hemos agregado ahí funciones... esos external storage providers tienen "como" folders que pueden ser privados o pueden ser públicos, hay también dos funciones distintas de acuerdo a como lo quieres tratar al archivo, si es un archivo público o si es un archivo privado. Si es un archivo privado, después te va a dar un link que también expira o tu le pides al external provider que te de un link para ese momento y que ese momento dure un minuto, dos minutos, una hora, lo que tu quieras, para después mostrar ese link seguramente en el browser y que el usuario pueda hacer click, pero que ese link ya deje de ser válido luego de un tiempo. External Storage y tipo de data ofile ahí al respecto.

Services

SOAP

OPEN API (Rest)

+

Services through Data Stores

ODATA

CRUD

Data View → Transaction → Attributes → for each

Bueno, y después agregamos Odata, que es un protocolo rest a través del cual otros proveedores exponen su modelo de datos como entidades y a esas entidades se les puede dar alta, bajas y modificaciones.

Lo interesante en GeneXus es que esos servicios se pueden importar con un inspector, se puede navegar, entonces las entidades de un determinado modelo Odata expuesto así, se impronta y cuando se impronta quedan en GeneXus como transacciones (data viwes) y queda un data store asociado a eso, después ustedes programan for each y ni se dan cuenta que en realidad están no haciendo un select sobre la base de datos, sino una llamada o un servicio para obtener esos datos, ahí vamos a entrar en detalle luego también.

Streams & Notifications apis

Sockets

Kafka

JPush, OneSignal

... y después hemos agregado Sockets y Kafka como mecanismos para comunicación bidireccional, mecanismos bidireccional Sockets, eso se usa mucho en aplicaciones de Chat, adentro de la misma aplicación.
Y Kafka se usa para comunicar distintas aplicaciones.
Se hicieron mejoras en OneSignal como mecanismo de envío de notificaciones.

Payment apis

MercadoPago

Paypal

Stripe

WeChat Pay

Ali Pay

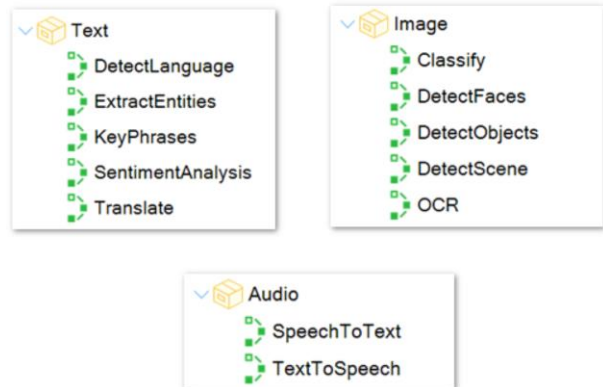
Después se agregaron apis de pago, y bueno, en cada región en las que ustedes están (estarán usando mas o menos algunas de estas apis) podemos agregar también nuevas apis, así como hemos agregado éstas para sus regiones si fuera necesario.

Artificial Intelligence

IBM Watson

Microsoft CS

SAP Leonardo


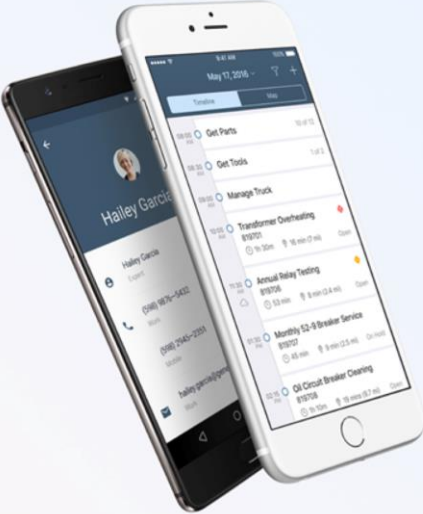


Con respecto a Inteligencia Artificial (AI) el primer paso que dimos en GeneXus es agregar métodos para analizar texto, analizar imágenes o procesar texto, imágenes y audios, por ejemplo para traducir, para sacarle la foto a una placa y obtener el número a partir de esa placa.

Y lo bueno en GeneXus es que ustedes tienen...esto ya funciona con tres proveedores y nosotros vamos a ir agregando nuevos proveedores, en estos momentos funciona para estos tres.

¿Cuál es la ventaja de que funcione para múltiples proveedores?

Bueno, primero, a ustedes seguramente quizás les convenga usar uno u otro proveedor de acuerdo a las... no se, políticas de la empresa, pero además ustedes pueden... quizás algún proveedor es más experto o mejor en alguna función y algún otro proveedor es mejor analizando otra función... quizás por decir algo, SAP sea bueno en SpeechToText, pero quizás IBM Watson sea el mejor para DetectFaces, no se si es así, es simplemente como un ejemplo ¿ok? Y entonces ustedes pueden (con el mismo código en GeneXus) pueden probar contra los distintos proveedores y quedarse con el que más les conviene.

Accelerate adoption of SAP Systems

- Automatically Extend SAP S/4HANA®
- Leverage SAP HANA DB,
- Deploy to SAP Cloud Platform
- Enjoy SAP Fiori UX
- Take advantage of SAP Leonardo

SAP Silver Partner

SAP® Certified
Integration with SAP® S/4HANA

Hemos creado GeneXus for SAP Systems también, como una edición mas de GeneXus 16. Ese GeneXus for SAP Systems tiene: bueno, pueden extender SAP ERP, o sea pueden inspeccionar los puntos de extensibilidad de SAP y entonces hacer aplicaciones que se integran con el propio ERP, pero después SAP también tiene un motor de base de datos, así que ustedes pueden usar SAP HANA DB como motor de base de datos, no el motor de base de datos del ERP sino como su propio motor de base de datos de su propia aplicación y esa aplicación la pueden subir después a la nube de SAP. Y como ya les decía, esa edición de GeneXus ya tiene un patterns incluido que sigue las guías de diseño de SAP. Entonces si van a hacer una aplicación para el mundo de SAP, lo mas natural es hacerla con ésta edición de SAP porque los usuarios que están utilizando SAP si empiezan ahora a utilizar la herramienta que ustedes le construyen (la solución que ustedes construyen) esa solución ya sigue las guías de interface, entonces no les tienen que explicar nada, ellos ya van a saber donde encontrar el botón y les va a resultar super natural trabajar con la solución que ustedes les hagan.

¿?

Eso es una edición especial de GeneXus.

Multi-experience & Omnichannel

Integration

Security

Build

Test



Data Security

Sensitive Data Protection

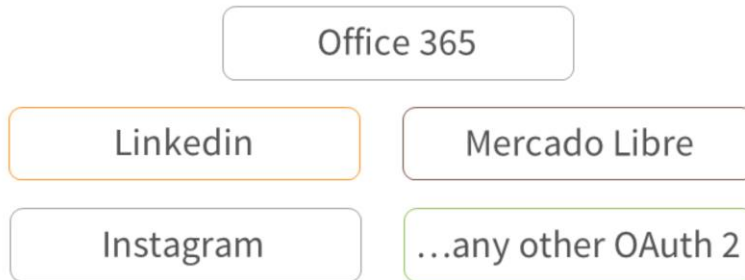
Apps Obfuscation

Offline DB Encryption

GeneXus Access Manager

Con respecto a seguridad (Security) hemos trabajado en la protección de datos, lo que es los datos sensibles en el web, las variables read only y demás. Y después en Smart Devices también la ofuscación de las aplicaciones y la encriptación de la base de datos Offline para lograr estándares de calidad y de seguridad requeridas en bancos.

OAuth 2 & SSO



Y después a nivel de GAM, hemos agregado la posibilidad de que ustedes puedan definir el GAM como un identity provider de cualquier solución que cumpla con Oauth 2.0, así que pueden integrarle con Office 365 o con el autenticador que viene con el "azure" que también sigue lo mismo o MercadoLibre, Instagram o cualquier otro.

Multi-experience & Omnichannel

Integration

Security

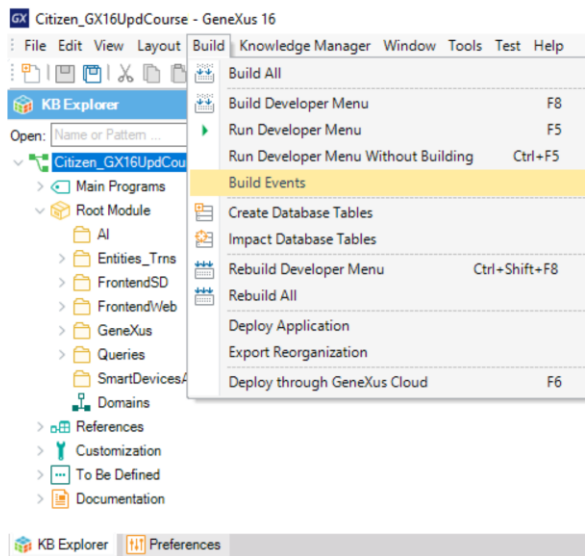
Build

Test



Build

Build events



Con respecto a Build, hay algo nuevo en GeneXus, “Build events” eso va en el menú de Build y hay una opción nueva, pero en este menú ustedes pueden definir que cosas hacer antes del Build y que cosas hacer después del Build, por ejemplo copia de archivos o ejecutar algún “bat” que haga un proceso extra. A veces ustedes precisan eso, es para usuarios avanzados normalmente esta funcionalidad, pero pueden definir ahí, las definiciones que ustedes pongan ahí quedan como una propiedad oculta a nivel de environment, o sea, el próximo “commit” de su environment pueden enviar eso al servidor, al GX Server, queda todo integrado, así que si en su equipo tienen que agregar tres archivos luego de cada Build y tomarlo de algún file systems, bueno, lo agregan aca y entonces el Build lo hace automáticamente.

New generators

.NET Core

Apple Watch

Apple TV

Bueno, además de los generadores que les hablaba (Apple Watch y Apple TV) el generador que agregamos es .NET Core, así como framework ahora generamos también para .NET Core con la misma licencia y es el mismo generador .NET...

.NET Core es la tendencia de MicroSoft, MicroSoft apunta a todo .NET Core, las versiones de .NET Core suceden mucho más rápido que el .NET framework, o sea, es evidente la movida hacia .NET Core.

.Net Core es mas rápido, es mas escalable y es mas liviano, y corre además en Linux, es multi-plataforma, entonces para subir aplicaciones también es atractivo por ellos y para correr en contenedores es la solución a utilizar, así que todo lo que puedan empiezan a moverse a .NET Core, esa es la primer versión de este generador... es decir, tenemos la primer versión de estos tres generadores, vamos a seguir mejorándolos, pero específicamente con .NET Core todavía no funciona con todos los DBMS, en general porque nos falta a nosotros trabajo de integración y de prueba, ya funciona con MySQL, Postgre y SQL server (me parece que estos son en los tres que funciona). Y tenemos en el Wiki escrito la lista de funcionalidades que no soportamos, así que ustedes pueden seguir una vez esas Checklist y entonces ver si aplica a su solución o no. Si pasa toda la checklist ya inténtenlo y generen.

Estaría bueno después tener feedbacks también de ustedes, en los puntos que faltan por si nosotros no nos dimos cuenta de algo, pero en realidad la lista es bastante completa, o sea, vamos a ir por los puntos puntos que faltan e irlos completando.

- En la parte NetWork digamos ¿podemos movernos fuertemente o lo tenemos que hacer con cuidado como si fuera un "beta"?
- Mira, no, es una primer versión... no es un beta, lo pasamos a la versión estable porque si te compila te va a funcionar básicamente, o sea, el código del .NET Core no es tan distinto con el .NET framework, sí tuvimos que hacer cosas con el código generado y hay ciertas funciones que sí, que si no usan otras librerías base, que si no, no te compila y esas librerías base las fuimos migrando, no se, todo lo que es manejo de "date time" por ejemplo y con "mili segundos" tuvimos que hacer una implementación específica, o sea, hay como

Splits, hay como códigos separados para algunos componentes sí, pero también hay una gran base de códigos en común, entre .NET framework y el "Beta".

Si hay algo que no hemos contemplado es probable que te surja ya en tiempo de compilación, así que es una etapa bastante temprana para darte cuenta que el proyecto no va a andar o saber que limitaciones tiene.. Y después, no se... testear, pero...

- La pregunta la hacia por dos partes, una por GeneXus pero la otra por el lado de MicroSoft...
- Bueno, MicroSoft ha puesto soluciones en producción con .NET Core y Azure, todo lo que tienen para poner en producción es con .NET Core, Amazon también, todo lo que se ve sobre .NET (soluciones .NET) en Amazon para correr en contenedores y demás también va todo sobre .NET Core, o sea... la base esa está bien... es simplemente que en algunos puntos faltan librerías y código a emigrar todavía.
- Gracias!

Multi-experience & Omnichannel

Integration

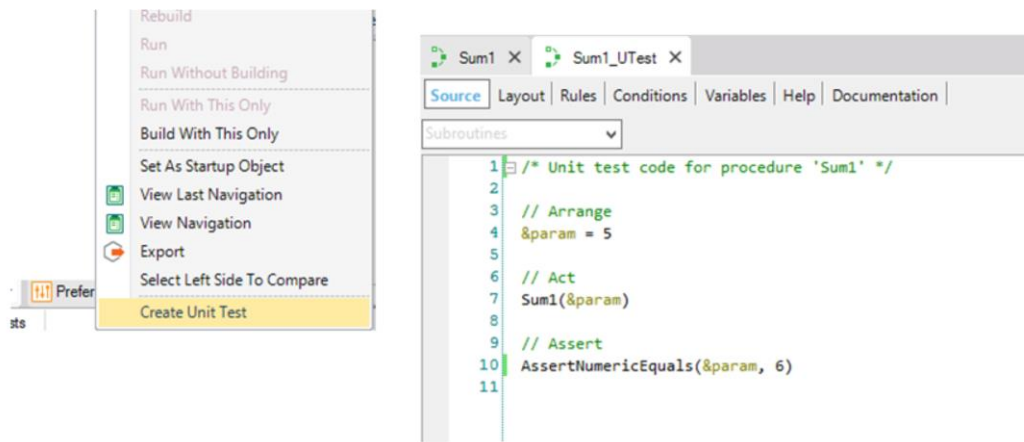
Security

Build

Test



Unit Testing



Y después, con respecto a testing, agregamos en GeneXus en el IDE, en el IDE en forma free que ustedes, para cualquier desarrollador que pueda crear nuevos Unit tests y que esos Unit tests los pueda ejecutar también y los pueda probar él mismo desde el IDE.

Así que esa funcionalidad fue desarrollada para abstracta pero viene Built in en el GeneXus y la pueden usar libremente, así que empiecen a crear Unit tests para testear automáticamente, por ejemplo procedimientos, Datas providers y Business Components.

Después Abstracta esta trabajando en una versión nueva de GX test, se llama GXTest 4 (creo que la quieren tener lista para Noviembre del 2018) y ese GXTest 4 viene también con integración con el IDE de GeneXus pero es para todo lo que es automatización del test de UI, de interfaz de usuario. Esto es para Test unitarios.

Operations

Package/Release/Deploy

Deployment Unit Object
 Build events
 Serverless (deploy to AWS lambda)
 GX Cloud Deployment Services → F6 PAAS
 Containers (Docker)

Configure/ Operate

ConfigurationManager API
 Config via Environment variables

Monitor

Log API



Bueno... y con respecto a operaciones hemos trabajado en el Deploy, agregando Deploy para ambientes serverless también, contenedores.

Y después tenemos unas APIs nueva, Configuration management Api es para leer el archivo de web.config o el client cfg. Con eso pueden leer todas las configuraciones que actualmente se están utilizando.

Y después... ustedes además... la configuración se puede leer a través de variables de ambiente, o sea, si las variables de ambiente que están activas en ese momento, en ese ambiente... tienen un nombre especial, GeneXus automáticamente lee la información a partir de esas variables de ambiente.

¿eso para que sirve? Bueno, quizás ustedes en el web.config o en el client cfg en desarrollo tienen ciertas configuraciones, pero en producción hay que configurar otras, entonces... antes de levantar el contenedor o antes de levantar el ambiente de producción simplemente en el ambiente de configuración se setean variables de entorno y ya se toman en cuenta esas sin tener que re empaquetar la solución con otros client cfg o sin tener que modificar ese web.config.

```
Event 'WriteAllLogLevels'  
  Composite  
    Log.Debug("Debug message","TestSDLoggingDebug")  
    Log.Info("Info message","TestSDLoggingInfo")  
    Log.Warning("Warning message","TestSDLoggingWarning")  
    Log.Error("Error message","TestSDLoggingError")  
  EndComposite  
Endevent
```



Y algo para mejorar el loggin o incluso después va a servir para mejorar el monitoreo, es que tenemos una API de log (perdón, aca esta escrito con variable debug pero en realidad es estático, escriben Log.Debug o Log.Info, Log.Warning o .Error) tienen distintos métodos asociados al tipo dato log, al External Object Log para grabar en el log.

Y en el .NET se graba en el client.log como siempre y en Java ahora también va a haber un client.log para grabar esto, o sea, en el .Log o en el lugar que ustedes especifiquen a través de unas propiedades, ustedes pueden gobernar con distintas propiedades a donde es que se va a grabar ese Log, si en la consola, si en un penLog o si se va a enviar a algún otro lado.

DevOps

GeneXus

Multi-experience & Omnichannel

Integration

Security

Build

Test

Package/Release/Deploy

Configure / Operate

Monitor



Pipeline control

Bueno, y después...

Cuando uno está en un ambiente y quiere integrar y liberar... o sea, integrar todo el código, normalmente tiene una maquina que esa maquina empieza a hacer Build y empieza a hacer la compilación y quizás la puesta en producción también, ahora todos esos pasos o en conjunto con el paso de test de alguna forma tienen que ser gobernados, no puede ser que simplemente haga Update, haga Build on y ya lo suba a producción, sino que si en el medio hay un paso de test y ese test falla, entonces no tengo que subir a producción. Todo ese control se hace con herramientas como Jenkins para el Pipeline control...

Pipeline control



Source Code Management

None

GeneXus Server ?

GeneXus Installation ?

GeneXus Server URL ?

GeneXus Server Credentials ?

Knowledge Base Name ?

Version ?

Subversion ?

... y tenemos integración con Jenkins en GeneXus 16.

En Jenkins especifican cual es el GeneXus server que van a utilizar y configuran ahí todo para que se pueda hacer.

DevOps

GeneXus

Multi-experience & Omnichannel

Integration

Security

Build

Test

Package/Release/Deploy

Configure / Operate

Monitor



GeneXus Server

BPM

} Suite

Y Bueno, además de GeneXus 16, obviamente que esta lista toda la suite de trabajo, está GeneXus server y BMP, cada uno con sus funcionalidades nuevas también (en las que ahora no entro en detalle) pero en GeneXus server se mejoró mucho la performance y algunas funcionalidades menores. Y en BMP se trabajó mucho en la escalabilidad de GXFlow, en la performance de GXFlow y que GXFlow pueda ejecutar también en ambiente de plataforma como servicio.

Create, **evolve** and **mantein**
unprecedented software
solutions **automatically**

Bueno, muy bien...

Eso es un overview de lo que hemos trabajado en GeneXus 16.

La idea es que hoy ustedes puedan desarrollar lo que tienen con GeneXus 16, pero también sabemos que, no se, si el sistema lo tienen que mantener el año que viene o el otro año, también van a precisar entonces un GeneXus para mantenerlo y no necesariamente el GeneXus 16 de ahora, porque quizás en dos o tres años haya otros protocolos que ahora ni siquiera conocemos, entonces estamos construyendo GeneXus de esa forma, que ustedes puedan también ir evolucionando con GeneXus sus mantenimientos, o sea la solución que hoy arrancaron quizás con GeneXus 15 quizás ahora la siguen desarrollando con GeneXus 16 y en unos años con GeneXus 17 y así...

Así que también tenemos mucho cuidado en la compatibilidad y en que ustedes puedan fácilmente adoptar las versiones porque justamente sabemos que es un trabajo continuo el desarrollar y el crear soluciones, esos son los principales puntos de GeneXus 16.

GeneXus™

The power of doing

More videos training.genexus.com
Documentation wiki.genexus.com
Certifications training.genexus.com/certifications