

Principales Características

PXTools



Torre de los Profesionales
Yaguarón 1407 Oficina 909
Montevideo – Uruguay

Tel: (598) 290 46 2 46
Email: ventas@puntoexe.com.uy



Patterns
Web & SD



Modules**Web**



Modules**SD**



User**Controls**



External**Objects**



Extensions



External**Programs**

OBJETIVO



Brindar al desarrollador GeneXus una serie de herramientas que le permita disminuir los tiempos de desarrollo.

Los principales elementos que la componen están organizados en categorías referida a distintos tipos de objetos del lenguaje de programación GeneXus. Cada uno de estos tipos de objetos permiten mejorar (de distinta manera) los tiempos de desarrollo:

- **Patrones:** Detección de patrones de comportamiento común de las aplicaciones y generación de automatismos de programación basados en esos patrones. Disminuye los tiempos ya que automatiza mucha programación, que por el contrario, tendría que realizarse a mano. Los patrones soportan distintos generadores. Los generadores terminarán creando distintos objetos GeneXus de acuerdo al concepto de plataformas.
- **Módulos:** Definición encapsulada de programas para cumplir con una funcionalidad específica. Disminuye los tiempos ya que predefine mucha programación, que por el contrario, tendría que desarrollarse por completo.
- **User Controls:** Controles de edición que no vienen predefinidos en GeneXus, que se pueden encontrar disponibles en la Web y que se integran al ambiente de desarrollo de los formularios de las interfaces gráficas de GeneXus. Disminuye los tiempos ya que facilita la integración con controles externos y disminuye los tiempos de investigación y desarrollo.
- **External Objects:** Funcionalidades programadas en alguna plataforma de lenguajes de programación soportados por GeneXus que son integrados en forma natural dentro de GeneXus. Disminuye los tiempos porque permite fácilmente incorporar funcionalidades que no están soportadas en GeneXus y ahorra mucho tiempo de investigación y desarrollo.
- **Extensiones:** Programas que se integran al ambiente de desarrollo GeneXus. Disminuyen los tiempos ya que incorpora funcionalidades en el ambiente de desarrollo que por el contrario tendría que realizarse a mano. Muchas de estas extensiones son para trabajar con gran cantidad de objetos para realizar búsquedas o transformaciones.
- **External Programs:** Programas (por lo general Command lines) que permiten algún grado de automatización de funcionalidades externas al ambiente de desarrollo.

PLATAFORMAS

Las plataformas soportadas son:

Plataforma Web Desktop

Los patterns relacionados con esta plataforma generarán distintos tipos de objetos GeneXus: WebPanels, Procedimientos, SDTs, Transacciones. Dependiendo de los patrones se generarán alguno de estos objetos mencionados.

La modalidad Web Desktop principalmente hace un diferencial en los WebPanel a desarrollar donde el Form generado para esos WebPanels son basados en HTML.

En esta plataforma se presentan en forma predeterminada diseños predefinidos basados en colores básicos que permiten fácilmente implementar un ambiente muy refinado. Estos diseños predefinidos vienen asociados a los controles GXUI que permiten entre otras cosas Controles muy avanzados para el trabajo en ambientes Desktop (Áreas colapsables y flotantes, Toolbars, Menús, TreeView, etc.). Los controles GXUI están basados en el producto Ext Js de la empresa Sencha (www.sencha.com) el cual es licenciable y deberán ser adquiridos en forma independiente del producto PXTools.

En caso de no querer utilizar los controles GXUI PXTools brinda un diseño base por el cual el programador junto con el diseñador web tendrán la libertad para definir la estética visual de forma totalmente libre.

Plataforma Web Responsive

Los patterns relacionados con esta plataforma generarán distintos tipos de objetos GeneXus: WebPanels, Procedimientos, SDTs, Transacciones. Dependiendo de los patrones se generarán alguno de estos objetos mencionados.

La modalidad Web Responsive principalmente hace un diferencial en los WebPanels y Transacciones a desarrollar donde el Form generado para esos objetos son basados en Abstract Layout (nuevo tipo de Form soportado a partir de GeneXus X Evolution 3 para soporte de ambiente responsivo).

Para acompañar este generador se está trabajando en el soporte de nuevos controles responsivos como ser: el Smart Menus (que genera menus horizontales y verticales) y el Responsive Layout (que genera side bars) todos ellos responsivos, que permitirá integrarlos a las masterpages y templates de PXTools para la representación de un ambiente funcional completo y prediseñado con el objetivo de facilitar al desarrollador la puesta inicial de los sistemas a desarrollar.

Plataforma Smart Devices

Los patterns relacionados con esta plataforma generarán distintos tipos de objetos GeneXus: Work with for SD, Panel for SD, Procedimientos, SDTs, Transacciones. Dependiendo de los patrones se generarán alguno de estos objetos mencionados.

CARACTERÍSTICAS

Las principales características de PXTools son las siguientes:

3 Plataformas

- Web Desktop
- Web Responsivo
- Smart Devices

13 Patrones

- Work With
- Parameter Request
- Composer
- Flow Controller
- OAV
- Synchronization
- Audit
- Report Template
- Entity Parameters
- WS Layout
- WS Transaction
- WS Query
- WS Data

2 External Programs

- XPZ Converter
- KB Synchronization

10 Extensiones

- Instance Find and Replace
- License Manager
- Install Manager
- Apply Default
- Instance Attacher
- Revert to Previous Date Version
- Change Reference
- Save Modules
- Save Objects
- Generate Web Startup Object

32 Módulos Web

- APIs
- Menus
- System
- Process Status
- Mail Accounts
- Send Mails
- Receive Mails
- System Parameters
- Alerts
- Batch Print
- Control Preferences
- Documents
- Dynamic Call References
- Projects
- Task Manager
- Audit Log
- OAV
- People Organizations
- Security
- Security Server
- Security Client
- Synchronization Log
- Synchronization Web Services
- Countries States Cities
- Printer Markup Language
- HTML to PDF
- File Storage
- Cloud Tasks
- Web Services Log
- Table Cleaner
- OAuth Service
- OAuth Customer

2 Módulos SD

- Help
- Debug

16 User Controls

Generales

- Password Quality Checker
- Scroll Line
- Password Flash Letter
- Password SD
- InConcert
- JSON Viewer
- Platform Detector
- Color Picker
- Javascripts

Responsivos

- Responsive Layout
- Smart Menus
- Grid Horizontal Scroll
- Grid Paging
- Contextual Box
- Smart Actions
- Code Scanner

14 External Objects

- Android Send SMS
- Android Bluetooth Print
- Android Bluetooth Receiver
- Android BLE Receiver
- Android XMLTools
- Java QRCode Generator
- Java XMLSignature
- Java XMLUtils
- Java File Properties
- Java Chart Generator
- Java IP Mask Validator
- Java SOAP Mutual TLS
- Java Image Util
- Java Shell Command

PATRONES



Patterns Web & SD

Los patrones tienen como objetivo generar objetos GeneXus que tienen un comportamiento repetitivo. Definen un modelo preestablecido y arbóreo que permita la definición de variaciones de dicho desarrollo y dispone de un generador que permita analizar las variaciones y generar un código final para cumplir la funcionalidad establecida por el patrón.

GeneXus brinda una plataforma base que facilita la creación de estos patrones.

Uno de los primeros objetivos por lo que nacieron nuestros patrones era establecer una plataforma de desarrollo que facilite el proceso de migración de aplicaciones Win y a Web, incorporando funcionalidades intrínsecas destinadas a resolver complejidades que implica el pasar de un modelo Win (con programación modal) a un desarrollo Web (donde no hay dependencia modal entre interfaces gráficas).

Usamos todo nuestro ingenio para lograr implementar interfaces gráficas Web con una simulación de código modal entre interfaces minimizando drásticamente la necesidad de un rediseño del sistema al migrar a web.

Otro de los objetivos iniciales del uso de patrones era simplificar la metodología de la programación de una interfaz logrando un modelo declarativo genérico que no dependa exclusivamente en la generación de HTML sino que el modelo declarativo fuera tan genérico que:

- El desarrollo de las interfaces sea simple para programadores Windows que recién estaban incursionando en el desarrollo Web y que no tenían conocimiento ninguno del uso del HTML.
- La declaración de la interfaz debía manejar conceptos tan genéricos que puedan entenderse hasta para elaborar formularios en un editor de texto manejando elementos como Row, Columns, Rectangle, Tabs que perfectamente pueden entenderse para el desarrollo de una interfaz no importando lo que tenga que generarse como objeto final.
- Nos permita no solo generar código HTML sobre lo declarado sino que nos podamos generar sobre cualquier otro modelo de interfaz diseñado por GeneXus logrando así minimizar el proceso de migración hacia nuevas tecnologías que GeneXus pueda implementar en el futuro.

Es así como hemos logrado que nuestros clientes hoy en día puedan desarrollar interfaces HTML como interfaces con Layout Abstracto en los Web Panels y Transacciones GeneXus y aún más nos permitió que un mismo desarrollo pueda implementarse a nivel del generador Smart Device de GeneXus.

Con el paso del tiempo el uso de patrones no solo cumple el objetivo de desarrollar interfaces gráficas sino todo tipo de desarrollo repetitivo.

Disponemos de años de desarrollo de sistemas de muy distintos tipos en donde hemos detectado la repetitividad en el desarrollo de programas y transformarlos en patrones para simplificar, agilizar y automatizar el desarrollo.

A continuación presentamos los distintos patrones que hemos desarrollado:

- **Work With (Web & SD):** Basado en el patrón Work With de GeneXus pero potenciado para permitir una gran cantidad de funcionalidades relacionadas con la manipulación de registros de tablas de la base de datos.
- **Parameter Request (Web & SD):** Soporte de ingreso de datos tabulares para invocación de procesos, reportes u otras interfaces gráficas. También sirve como forma de presentar datos tabulares.
- **Composer (Web):** Permite la integración de componentes de interfaz gráfica desarrollados por otros patrones logrando así pantallas complejas con interacción de múltiples grillas y secciones tabulares.
- **Flow Controller (Web):** Permite la declaración de la interacción entre distintas interfaces gráficas, simulando el concepto de diálogo modal. Es totalmente complementario del GXFlow de GeneXus ya que simplifica los diseños del modelador.
- **OAV (Web & SD):** Las siglas significan Object Attribute Values. Permite la definición de atributos en run-time sobre una entidad definida en design-time. Ahora también orientado al soporte de encuestas (interdependencias de preguntas y relacionamiento con una entidad cualquiera del sistema)
- **Audit (todas las plataformas):** Soporte de log de auditoría generando reglas en la transacción donde almacenan la información en el módulo Audit. También está soportada la definición de auditoría de registros (definiendo atributos estándares para guardar información como: usuario última modificación, fecha-hora de última modificación, etc.). Es posible auditar procesos que modifiquen las tablas del sistema utilizando las transacciones como Business Components.
- **Synchronization (todas las plataformas):** Sincronización de tablas entre base de datos de dos puntos que pueden ser de distintas plataformas: Web, Win, Pocket PC, Smart Devices y distintos servidores de bases de datos.
- **Report Template (Win & Web):** Generación estándar de Header y Footer en los reportes GeneXus. Basado en declaraciones de Templates que permite adaptar todos los reportes de la KB con mínimo esfuerzo.
- **Entity Parameters (Web):** Permite definir parámetros asociados a una entidad del sistema. La principal diferencia con el OAV es que el Entity Parameters permite manejar parámetros definidos en tiempo de diseño y los valores se definen en tiempo de ejecución mientras que el OAV permite definir atributos y valores ambos en tiempo de ejecución.
El que el parámetro sea definido en tiempo de diseño permite un mayor uso dentro del sistema para afectar el comportamiento del sistema en función de los valores de dichos parámetros.

- **Patrones de APIs (Web):** Los cuatro patrones de Web Services permiten modelar en forma completa un set de APIs para permitir que entidades externas al sistema tengan integración con el mismo. Para ello se modelaron 4 tipos de patrones para conformar el universo de APIs necesarias para soportar esta integración:
 - a. **WS Query:** Este patrón tiene como objetivo devolver una colección de registros asociado a una tabla del sistema.
Permite la definición de posibles atributos de la tabla que conformarán filtros a realizar sobre la tabla principal así como también posibles órdenes a definir para la recorrida de la tabla.
 - b. **WS Data:** Este patrón permite recuperar la totalidad de atributos pertenecientes a un registro de una tabla.
Estos atributos pueden ser propios de la tabla o pertenecientes a la tabla extendida.
 - c. **WS Transaction:** Este patrón permite los procesos de Ingreso, Modificación y Borrado de registros de una tabla.
Este patrón quedará asociado a una Transacción para trabajar en modalidad de Business Component y por consiguiente mantener todas las reglas de negocio necesarias para mantener coherencia de la estructura de las mismas.
 - d. **WS Layer:** Este patrón es el que publicará los servicios a nivel público tanto como servicios SOAP como REST.
También permite establecer entradas que permitan interactuar con procedimientos internos de la base de conocimiento incorporando en todos los casos los niveles de seguridad que uno quiera implementar.

MÓDULOS PARA WEB



Los módulos PXTools son un conjunto de objetos GeneXus que definen un modelo completo de gestión encapsulado una funcionalidad concreta que será posible incorporar como parte de desarrollo de un sistema.

El objetivo de estos módulos es definir una caja negra estándar para la gestión de una funcionalidad específica que usualmente es necesario implementar en el proceso de desarrollo de cualquier sistema de software.

Su programación permite la interoperabilidad entre distintos módulos de PXTools ya que en ocasiones hay dependencias necesarias para hacer uso de un módulo.

A continuación presentamos los distintos módulos desarrollados destinado al uso de sistemas web:

- **APIs:** Programas básicos para el funcionamiento correcto de los patterns PXTools.
- **Menus:** Diseño de menús con soporte de distintas formas de visualización (Tabs, Toolbar, Tree-View, Image Bar, standard o combinaciones). Soporte de favoritos y búsquedas sobre lateral izquierdo. Seguridad integrada y automática. Soporte de Image Bar como menú principal o como menú de acceso directo.
- **System:** Módulo que contiene algunas entidades utilizadas por varios módulos (Declaración de Objetos, Declaración de Módulos del sistema, etc).
- **Process Status:** Realiza el seguimiento y control de ejecución de procesos. Control de concurrencia y posibilidad de interrumpir el proceso sometido. Posibilidad de enviar mail al usuario asociado a la ejecución para informar la finalización del proceso. Soporta actualmente la cancelación de procesos batch Command Line sobre la plataforma linux.
- **Mail Accounts:** Módulo para la gestión de Cuentas de EMail que podrá ser utilizada (opcionalmente) en el módulo de Send Mails y que deberá ser utilizada (obligatoriamente) en el módulo de Receive Mails. Permite realizar la verificación de las credenciales de acceso al momento de la declaración para minimizar problemas de conectividad al momento de utilizar los módulos de envío y recepción de Mails.
- **Send Mails:** Módulo para el envío de mails en formato texto o HTML. Soporte de conectividad con servidores SMTP con o sin autenticación. Soporte de conectividad vía POP3 en caso de no haber autenticación directa por el SMTP. Soporte de campañas de mail realizando tandas ordenadas por el FQDN (Fully Qualified Domain Name) para minimizar el tráfico a los distintos SMTPs.

- **Receive Mails:** Módulo para el control de recepción de mails al sistema. Definición de las posibles casillas de correo de recepción de información y conectividad con POP3 para leer los mails de las mismas.
- **System Parameters:** Modelo de declaración de parámetros del sistema sin necesidad de impactar la base de datos. Soporta distintos controles de edición (Character, Numérico, Fecha, Password, Memo, HTML o Combo). Configurable si el parámetro es multi-idioma. Soporte de grupos de parámetros por categoría.
La declaración de Combos se hace sobre Data Providers GeneXus lo que permite la muestra de valores de cualquier tabla del sistema.
- **Alerts:** Módulo para la generación de alertas por el sistema o a través de envío de mails. Soporte de seguimiento de una alerta hasta el momento que el usuario toma acción por la misma. Los destinatarios de las alertas pueden ser usuarios o roles del módulo de seguridad.
- **Batch Print:** Módulo para soporte de impresión batch. Definición de impresoras, estaciones y colas de impresión. Soporte de conectividad LAN (conectando directo contra la base de datos) o WAN (vía Web Services).
- **Control Preferences:** Módulo para el almacenamiento de las preferencias de los usuarios sobre los distintos controles GXUI en una tabla del sistema.
- **Documents:** Gestión de documentos basados en etiquetas. Soporte de versiones y declaración de cada página del documento integrando etiquetas que luego (utilizando ciertas APIs) serán sustituidas por los valores correspondientes generando finalmente un documento pdf.
- **Dynamic Call References:** Módulo encargado de almacenar todos los procedimientos GeneXus o interfaces gráficas que serán llamados en forma dinámica dentro de GeneXus. Definido especialmente para permitir que los usuarios puedan customizar el sistema sin tener conocimientos técnicos avanzados de los programas existentes en el sistema.
- **Projects:** Módulo para la interacción con proyectos. Está vinculado con el Login y tiene algunas pantallas de gestión de proyectos.
- **Task Manager:** Módulo encargado de la gestión y la ejecución de procesos batch del sistema. Soporte de tareas cíclicas y repetitivas.
Soporte de fallos de ejecuciones y posibilidad de reintentos de ejecución.
Soporte de tareas subordinadas de otras tareas.
También maneja hoy en día la gestión de Colas de ejecución lo que permite tanto la ejecución sincrónica como asincrónica de procesos.
- **Audit Log:** El log de auditoría de transacciones permite realizar un seguimiento de los valores de los atributos que se modifiquen a través de una transacción (sea por el form o a través del business component).
- **OAV:** Módulo encargado de dar soporte al patrón OAV. Contiene entidades predefinidas como la de Atributos y las APIs que serán invocadas por los objetos generados por el pattern.

- **People & Organizations:** Módulo para la representación de Personas y Organizaciones y toda la información común a ambas entidades (documentos, teléfonos, direcciones físicas, direcciones electrónicas, relaciones empleado-empleador y relaciones genéricas). Basado en el modelo presentado por Nicolás Jodal, mejorado y habilitado para el uso con PXTools.
- **Security:** Módulo de seguridad basado en Usuarios, Roles, Pantallas y Acciones. Soporte de definición de seguridad ad-hoc sin necesidad de predefinir la base de pantallas y acciones del sistema. Posibilidad de definición de acciones virtuales para controlar funcionalidades que pueden ejecutarse bajo situaciones específicas.
- **Security Connector:** Es un grupo de Objetos GeneXus que vienen a ser parte de las APIs de PXTools que son customizadas para utilizar el módulo Security de PXTools.
- **Security Silent Sign On:** Es un pequeño módulo que está diseñado para que sistemas externos puedan ingresar al nuestro sistema sin requerir identificación de usuario de sesión.
- **Security Server:** Es una extensión del Módulo Security con el objetivo de establecer unas APIs de Web Services para establecer comunicación de la gestión de Usuarios con otras KB. Es la base para establecer un modelo de Single Sign On entre distintas Bases de Conocimiento.
- **Security Client:** Es una versión alternativa del módulo Security para definir seguridad en Bases de Conocimiento que establecerán un Single Sign On o un Silent Sign On contra una Base de Conocimiento principal.
- **Synchronization Log:** Módulo encargado de dar soporte al patrón Synchronization. Programación del visualizador del log de sincronización y de las APIs que serán invocadas por los objetos generados por el pattern.
- **Synchronization Web Services:** Predefinición de la capa de conectividad de los procesos de sincronización vía WebServices. Contempla el caso típico de replicación de un servidor central a distintos dispositivos remotos (vendedores). Almacenamiento tanto de los logs generados por las importaciones de cada dispositivo como las del propio servidor en tablas de este módulo que podrán ser accedidas, desde el backoffice, por técnicos especializados en caso de ser necesario.
- **Countries States Cities:** Módulo básico para la representación de países, estados y ciudades.
- **Printer Markup Language:** Módulo encargado de traducir el lenguaje de programación (declarado por PuntoExe) para el soporte de impresoras de tickets a los códigos de las distintas impresoras que emiten tickets. El objetivo de este módulo es actuar como un driver de impresoras de tickets. Por el momento este driver está soportando impresoras Datamax-Oneil APEX2 ,APEX3 y APEX4.

- **Statistics:** Pequeño módulo encargado de almacenar indicadores que se procesan generalmente por procesos batch que luego podrán usarse para distintos objetivos como ser simplemente desplegar dashboards o gráficas con la evolución de los mismos.
- **HTML to PDF:** Módulo encargado de la conversión de cualquier HTML a PDF. Interactúa con un software libre que permite esta funcionalidad.
- **File Storage:** Módulo para la gestión de archivos. La primera versión tendrá la gestión de documentos almacenados en la base de datos como blobs pero tiene el objetivo de concentrar la información de referencias de archivos en cualquier medio de almacenamiento de archivos.
- **Cloud Tasks:** Módulo para control básico de infraestructura. Puede controlar existencia de archivos o directorios en el File System, verificación de tareas programadas en el Cron de linux, gestión de Certificados Digitales y control de límite de tiempo de procesos batch para matarlos en caso de exceso de tiempos de ejecución. En caso de detectar alguna falla se informa a través del módulo de alertas.
- **Web Services Log:** Módulo para la gestión de logs de conexiones de Web Services. Está diseñado para mostrar los parámetros de entrada y su correspondiente respuesta de forma de garantizar principalmente al área de soporte la fidelidad de los datos recibidos y devueltos por cada servicio.
- **Table Cleaner:** Módulo destinado a la gestión de depuración de registros de tablas. Permite la determinación de tiempos (en años, meses o días) de permanencia de los registros. Utiliza el módulo de Task Manager para la ejecución de la depuración de los registros en batch.
- **OAuth Customer:** Módulo que permite la gestión de Tokens otorgados por sistemas externos que requieran una integración basado en el protocolo OAuth 2.0.
- **OAuth Service:** Módulo que permite brindar un mecanismo de autenticación a sistemas externos que se quieran integrar a nuestro sistema basado en el protocolo OAuth 2.0.
- **Web Service Layer:** Módulo que da soporte base a los 4 patrones (WS Layer, WS Query, WS Data, WS Transaction) asociados a la publicación de servicios web SOAP o REST asociados tanto a la consulta como a la actualización de datos de un sistema.

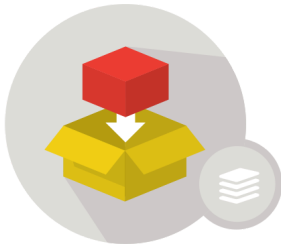
MÓDULOS PARA SMART DEVICES



ModulesSD

- **Help:** Módulo para gestionar todo el manejo de ayuda en línea. Soporte de conectividad Off-Line. Basado en uso de Wizard para guiar al usuario en las distintas pantallas. Posibilidad de invocar a la ayuda directamente desde las pantallas de ejecución posicionándose en el paso del wizard que corresponde con la pantalla que solicitó la ayuda. Declaración de la ayuda completa desde un Data Provider.
- **Debug:** Módulo para generar mensajes de debug por base de datos y pantalla de visualización de los logs.

EXTERNAL OBJECTS



ExternalObjects

- **Android SMS Sender:** Soporta el envío masivo de mensajes a teléfonos celulares.
- **Android Bluetooth Print:** Soporta el envío de texto a una impresora conectada por bluetooth. Se solicita la selección del dispositivo a utilizar en la primera impresión y luego queda guardada esa información durante todo el proceso que esté la aplicación abierta.
- **Java QRCode Generator:** External Object para lenguaje de programación Java para generar QRCode sin interacción con servicios de internet.
- **Java XMLSignature:** Herramienta para la firma de tags de XML.
- **Java XMLUtils:** Herramientas relacionadas con manipulación de XML para verificación de firma, verificación de schema de XML contra XSD y encriptación de elementos XML.
- **Java File Properties:** External object que permite leer archivos de texto para la declaración de parametrización del sistema.
- **Java Chart Generator:** Encargado de la interacción con la librería jfreechart de java para la generación de gráficas.
- **Java Linux Command:** Permite la ejecución de comandos Linux pero a diferencia del Shell de GeneXus permite ejecución de comandos Linux avanzados (por ejemplo hacer redirección de los resultados a archivos) o cualquier comando que se pueda ejecutar en una consola ssh.

USER CONTROLS



UserControls

- **Password Quality Checker:** Control de ingreso de password que permite ver la calidad de la password en 5 categorías.
- **Scroll Line:** Control que permite recorrer una línea de contenidos que supera el ancho del contenedor a través de dos botones de scroll horizontal.
- **Password Flash Letter:** Control de ingreso de password que permite mostrar la última letra ingresada en una zona (declarada por el usuario) con un tamaño de font que la destaca.
- **Password SD:** Control de ingreso de password imitando el usado en dispositivos Smart Devices.
- **InConcert:** Control que permite ejecutar comandos que interactúan con el sistema InConcert.
- **Responsive Layout:** Control para representar 5 regiones (Top, Left, Center, Right y Bottom) para la gestión de contenidos dentro de ellas. Soporta la customización de comportamientos en función de: Tamaños bootstrap (Extra Small, Small, Medium, Large) y Tipo de Dispositivo (Desktop o Mobile). Soporte de regiones Colapsables/Expandibles (Push) o Flotantes (Float). Soporte de región superior en modo inteligente (Smart) que pasa a tamaño minimizado al comenzar a realizar un movimiento ascendente en la pantalla.
- **Smart Menus:** Control para representar menús adaptativos según dispositivo.
Soporta 4 tipos de representación:
 - Horizontal Responsive Push: Representación horizontal responsivo por un límite de ancho y pasaje a menú vertical expandido.
 - Horizontal Responsive Float: Representación horizontal responsivo por un límite de ancho y pasaje a menú vertical flotante.
 - Horizontal Smart: Representación horizontal con pasaje automático a menú vertical flotante según el espacio que se disponga.
 - Vertical: Representación vertical expandida.
- **JSON Viewer:** Control para visualizar el formato de un JSON con colores y colapsamiento y expansión de los subelementos.
- **Platform Detector:** Control para informar la información de la plataforma en base a dos valores:
 - Dimensiones: Valores basados en Bootstrap para definir los tamaños del dispositivo. Los valores son: Extra Small, Small, Medium, Large.
 - Tipo de Dispositivo: Desktop, Mobile.
- **Javascripts:** Soporte de variados métodos Javascripts donde algunos de ellos interactúan con JQuery y JQueryUI principalmente para manipulación de elementos del DOM.

EXTENSIONES



Extensions

- **Instance Find And Replace:** Permite la búsqueda de elementos específicos en instancias de pattern. Permite también el reemplazo masivo.
- **License Manager:** Extensión encargada de gestionar el licenciamiento del producto PXTools.
- **Install Manager:** Permite la generación de una instalación para proyectos basados en el generador CSharp. Guarda la información de los distintos DataStores definido en el modelo para solicitar, durante el proceso de instalación, la identificación de cada una de las distintas bases de datos. También permite guardar cada una de las reorganizaciones hechas en un modelo e incluirlas en el instalador para correrlas una tras otra durante el proceso de ejecución de la instalación.
- **Apply Default:** Encargado de identificar aquellos objetos de la KB que tienen pérdida de dinamismo respecto a su patrón generador. Permite también la recuperación masiva de dinamismo.
- **Instance Attacher:** Extensión que permite la asociación masiva de patrones a distintos objetos del sistema.
- **Revert To Previous Date Version:** Permite la recuperación masiva de objetos a una versión previa a una fecha. Muy útil cuando no se realizaron frizados que permitan volver a un estado estable de la KB.
- **Change Reference:** Extensión que permite unificar distintos objetos (con un mismo nombre en distintos módulos) a un solo objeto. Cambia las referencias de todos los objetos.
- **Save Modules:** Encargado de detectar todos los módulos en la KB y guardarlos en una tabla del sistema.
- **Save Objects:** Encargado de detectar todas las pantallas de la KB y guardarlas en una tabla del sistema.
- **Generate Web Startup Object:** Genera un WebPanel GeneXus con el objetivo de poder declararlo como StartUp Object. Simplifica los procesos de migración Win a Web para que GeneXus no interactúe con los objetos de otras plataformas.

EXTERNAL PROGRAMS



ExternalPrograms

- **XPZ Converter:** Permite la conversión de XPZs de versiones viejas de GeneXus para hacerlos compatibles con las nuevas versiones. Basado en la declaración de Expresiones Regulares para su corrección o descarte para simplificar el proceso de importación en la última versión de GeneXus. Muy útil si se mantienen versiones de un producto tanto en versiones viejas como en versiones nuevas de GeneXus.

- **KB Synchronization:** Programa encargado de automatizar exportaciones e importaciones de KBs de GeneXus en versiones X Evolution 2 o superior. Esta herramienta hoy en día es utilizada para mantener actualizadas y en forma automática cada una de las KBs de PXTools correspondientes a los Módulos. Se permiten los mismos esquemas de declaración como está soportado a nivel del Development Environment.

En la exportación es posible indicar Folders o Módulos (e incluir todos los objetos que estén debajo de ella) ubicar objetos que comiencen, contengan o finalicen con determinado valor y también es posible excluir objetos del resultado de la selección.

En la importación es posible excluir objetos que vengan del xpz.

Es posible solicitar la creación de la KB en caso de no existir.