

Factory Soft Venezuela C.A.

Manual de Programación de Eventos

Versión 1.1

26 de junio de 2013

Historial de Cambios

Versión	Fecha	Observaciones	Responsable
V1.0	26/JUN/2013	Creación del Manual.	RJG
V1.1	27/JUN/2013	Se agregaron ejemplos de aplicación de eventos.	RJG

Contenido

Historial de Cambios	2
Funcionamiento General de los Eventos.....	4
El Administrador de Eventos	5
El Lenguaje Interpretado Factory	7
Objetos Globales Disponibles.....	7
Vista del Registro/Documento.....	8
Opciones Globales y de Usuario	9
Objeto de Datos de Ejecución.....	10
Datos Básicos del Evento.....	11
Datos y Condición de Ejecución	12
Envío de Correo.....	13
Envío de Avisos	14
Ejecución SQL Arbitraria	15
Información del Gancho y Formulario Asociado	16
Repeticiones de Alertas de Tiempo.....	17
Ejemplos de Eventos.....	18
Ejemplo #1: Validación de descuentos antes de guardar	18
Ejemplo #2: Validación de Ganancia Antes de Confirmar	21
Otros Ejemplos de Aplicación.....	24

Funcionamiento General de los Eventos

eFactory provee varias características que permiten personalizar su comportamiento, entre ellas están los Eventos.

Los eventos en eFactory son unidades programadas que pueden ser ejecutadas en algún momento del proceso de trabajo normal, con los cuales es posible ejecutar una tarea, como enviar automáticamente un correo, actualizar un valor por base de datos, o impedir que un documento se guarde bajo ciertas condiciones.

Los "momentos" en los cuales se puede ejecutar un evento son llamados "ganchos" cuando están asociados a alguna funcionalidad de un formulario: al guardar un documento, al confirmarlo, al eliminar un registro, etc.

Por ejemplo, en el Sistema Administrativo se puede programar un evento en el gancho "antes de guardar nuevo" del documento Facturas de Venta, para impedir que se guarde una factura con un margen de ganancia por debajo de un porcentaje mínimo.

También puede ejecutarse un evento bajo ciertos parámetros temporales, como una fecha específica, o un intervalo regular. Los eventos basados en tiempo son llamados "alertas de Tiempos", y estarán disponibles en una próxima versión de eFactory.

La programación y configuración de los eventos comienza activándolos mediante las opciones "ACTEVE: Activar funcionalidades por Eventos" y "ACTALE: Activar funcionalidades por Alertas" (opciones de tipo lógico, módulo de Sistemas, Imagen 1) en la empresa y sistema (Administrativo, Contabilidad o Nómina) donde se deseen usar.

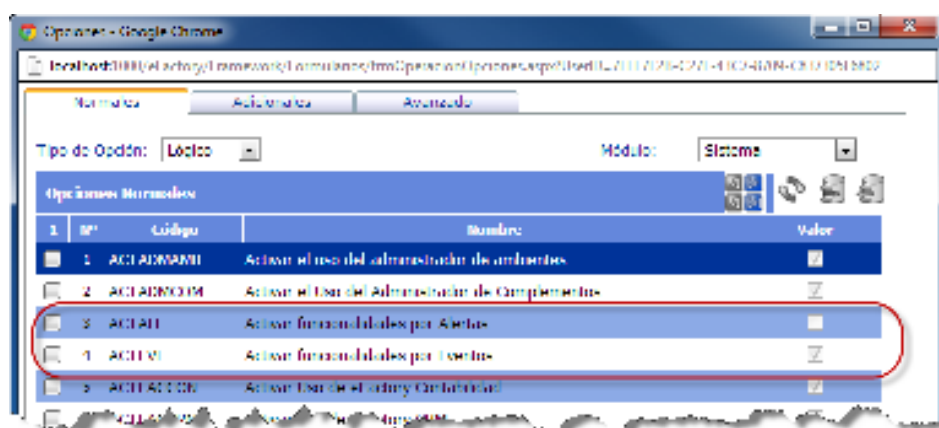


Imagen 1: Opciones para Activación de Eventos

El Administrador de Eventos

Los eventos se crean y asocian a un gancho por medio del Administrador de Eventos, disponible desde la ficha Herramientas de la ventana principal del sistema (Imagen 2).

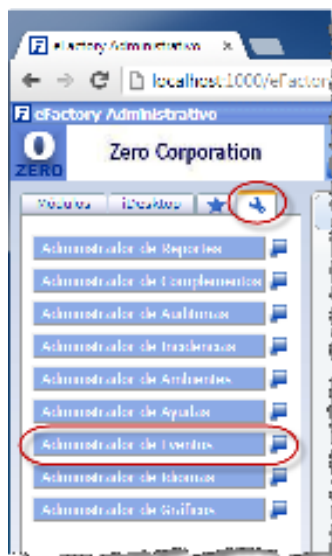


Imagen 2: Acceso al Administrador de Eventos

Al crear un evento se debe especificar la siguiente información (ver Imagen 3):

- Código y nombre del evento
- Tipo de disparador (Evento o Alerta)
- Datos y condición de ejecución
- Información para envío de correo, si aplica.
- Información para envío de avisos, si aplica.
- Sentencias SQL arbitrarias para ejecutar, si aplica.
- Información del formulario y gancho asociado, si el disparador es de tipo Evento.
- Condición para permitir o bloquear la operación (e.g. impedir que se guarde el documento), junto con un mensaje al usuario en caso de bloquear, si el disparador es de tipo Evento.
- Información de tiempo de ejecución y repeticiones, si el disparador es de tipo Alerta.

Parte de esta información consiste en datos básicos para el evento, sin embargo los campos más importantes contendrán código SQL o Código LIF (descrito en la próxima sección) que definirán el comportamiento y ejecución del evento.

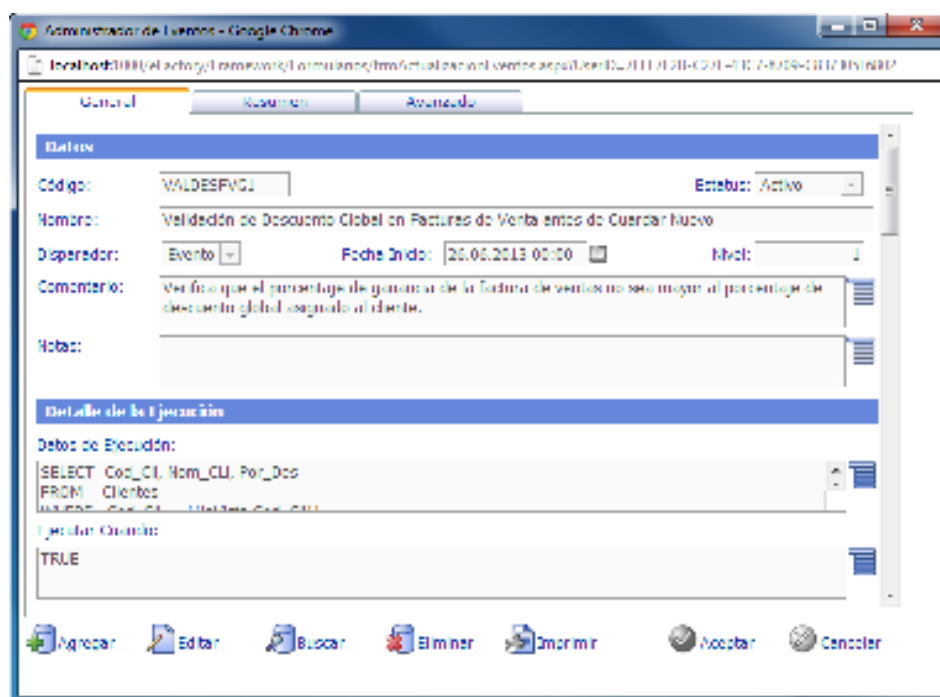


Imagen 3: Administrador de Eventos

Los campos que almacenan código SQL o código LIF, así como los campos Comentarios y Notas, disponen de una ventana de edición disponible haciendo clic en el botón a la derecha de cada campo mientras se esté editando el Evento (Imagen 4).

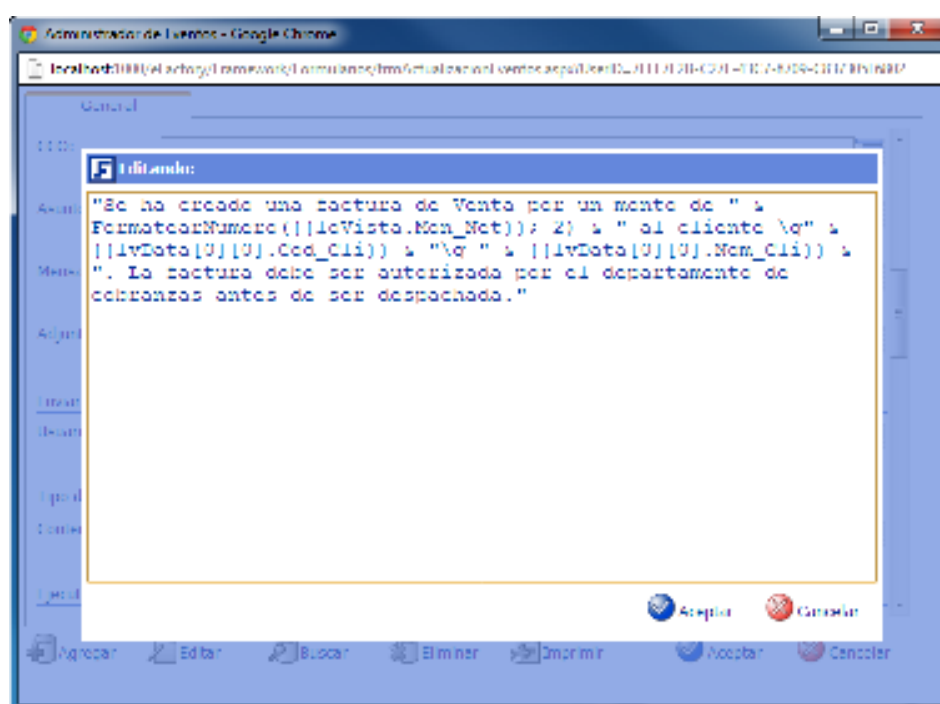


Imagen 4: Ventana de Editor Integrado

Si se hace clic en el mismo botón cuando no se esté editando el evento, entonces se tendrá acceso a un panel de solo lectura, que permitirá inspeccionar el contenido del campo cómodamente.

El Lenguaje Interpretado Factory

En esta sección se describirán algunas generalidades de Lenguaje Interpretado Factory (LIF) y algunos aspectos particulares de la implementación de LIF para programación de eventos.

LIF fue desarrollado con la finalidad de proveer una sintaxis simplificada de expresiones para dar soporte a diversas funcionalidades de eFactory, con el fin de facilitar la personalización de su funcionamiento.

Para mayor información sobre la sintaxis, operadores y funciones LIF refiérase al documento "Manual de Expresiones LIF".

Como adición a las características provistas por LIF, los diferentes componentes de eFactory proveen de características adicionales específicas de cada área. En los eventos, por ejemplo, se tiene acceso a los siguientes elementos:

- Objetos Globales: que proveen información del entorno en el momento de ejecutar el evento (datos de la empresa actual, de la sucursal, del usuario...)
- Vista del Registro/Documento: que provee información del registro o documento actual desde el cual se generó el evento. Solo está disponible si el evento es de tipo "Evento".
- Opciones Globales y de Usuario: Que permiten leer el estado actual de cualquier opción del sistema, sea esta una opción global, o una opción específica de algún usuario.

Objetos Globales Disponibles

goEmpresa: Hace referencia a la empresa activa al momento de ejecutarse el evento.	goEmpresa.pcCodigo goEmpresa.pcNombre goEmpresa.pcMonedaBase goEmpresa.pcMonedaAdicional
goSucursal: Hace referencia a la sucursal activa al momento de ejecutarse el evento.	goSucursal.pcCodigo goSucursal.pcNombre goSucursal.pnNivel
goUsuario: Hace referencia al usuario activo al momento de ejecutarse el evento.	goUsuario.pcCodigo goUsuario.pcNombre goUsuario.pnNivel goUsuario.pcCorreo goUsuario.pcSeparadorDecimal goUsuario.pcSeparadorDeMiles goUsuario.pcSeparadorDeFecha goUsuario.pcFormatoDeFecha
goEvento: Hace referencia a evento que se está ejecutando.	goEvento.pcCodigo goEvento.pcNombre goEvento.pnNivel goEvento.pcComentario goEvento.pcNotas
goAuditoria: Almacena información adicional utilizada para guardar auditorias del evento que se está ejecutando.	goAuditoria.pcNombreEquipo goAuditoria.pcDetalleVacio

Vista del Registro/Documento

Se dispone de dos objetos que dan acceso a los campos del registro o encabezado, y de los renglones del documento, si los hay.

loVista: Hace referencia los campos del registro activo, o del encabezado del documento activo, al momento de ejecutarse el evento (de tipo "Evento"). Las referencias al objeto loVista usan la siguiente sintaxis:

```
{{loVista.CAMPO}}
```

Donde `CAMPO` es el nombre de un campo del registro actual (o del encabezado, si es un documento). Por ejemplo, para leer el campo Código de Cliente de un documento de venta, se usaría:

```
{{loVista.Cod_Cli}}
```

laRenglones: Hace referencia los campos de los renglones del documento activo al momento de ejecutarse el evento (de tipo "Evento"). Si el registro activo no es tiene renglones entonces el objeto no estará disponible. Las referencias al objeto loRenglones usan la siguiente sintaxis:

```
{{laRenglones[M][N].CAMPO}}
```

Donde "`M`" es el número del grupo de renglones (el primer grupo de renglones será 0, el segundo 1, etc.). "`N`" es el número de renglón a leer (comenzando por 0 para el primer renglón). Y "`CAMPO`" es el nombre de un campo del renglón indicado por "`M`" y "`N`".

Normalmente, leeremos los renglones del documento en el campo "Datos de Ejecución" (en la próxima sección) y utilizando un bloque MACRO para recorrer todos los renglones como sigue:

```
--Tabla con los Artículos del lso renglones del documento (tomado de la vista)
CREATE TABLE #tmpRenglones(Renglon INT, Cod_Art CHAR(30))

#MACRO:laRenglones[0]#
INSERT INTO #tmpRenglones(Renglon, Cod_Art)
VALUES ({{laRenglones[0][N].Renglon}}, {{laRenglones[0][N].Cod_Art}})
#MACRO#
```

De esta forma, el bloque `MACRO` (LIF) generará una sentencia `INSERT` (SQL) por cada renglón del documento. El código SQL resultante será el finalmente ejecutado para generar los datos de ejecución. En la sección "Ejemplos de Eventos" se puede ver ejemplos de aplicación más completos.

Opciones Globales y de Usuario

Las opciones globales son aquellas creadas desde la pantalla de Opciones, en el módulo de Sistemas (Imagen 5).

1	OP	Código	Nombre	Valor
1	AA50000000		Activar asociación de Requerimientos Internos y Presupuestos	☐
2	AA00000000		Activar Descuentos Globales por Proveedor	☒
3	CO00000000		Confirmar Ordenes de Compra Automáticamente	☒
4	CO00000000		Confirmar Ordenes de Compras Automáticamente	☐
5	CO00000000		Confirmar Pagos Automáticamente	☐
6	CO00000000		Confirmar Presupuestos de Compras Automáticamente	☐
7	CO00000000		Confirmar Requerimientos de Compras Automáticamente	☐
8	IMP00000000		Solo imprimir Documentos con Pendientes en Ventas	☐
9	CO00000000		Activar Carga de Títulos de Entrada en Compras y Proveedores	☒
10	CO00000000		Activar Carga de Títulos de Entrada en Títulos de Requerimientos	☒

Imagen 5: Formulario de Opciones Globales

Las opciones de usuarios son aquellas creadas desde la pantalla de Opciones del Usuario, en el formulario de Usuarios Globales, ficha Avanzados (Imagen 6).

1	OP	Código	Nombre	Valor
1	CO00000000		Crear Usuario: Contraseña por Defecto	Mostrar
2	CO00000000		Crear Usuario: Nombre por Defecto	Mostrar
3	CO00000000		Crear Usuario: Email por Defecto	Mostrar
4	CO00000000		Crear Usuario: Puesto del Usuario de correo electrónico por Defecto	Mostrar
5	CO00000000		Crear Usuario: Puesto del Usuario de correo electrónico por Defecto	Mostrar
6	IMP00000000		Valor Especial al momento de	Mostrar
7	VAL00000000		Valor del tipo de registro de Registro en Almacenes	Mostrar
8	VAL00000000		Valor del tipo de registro de Registro en Almacenes	Mostrar
9	VAL00000000		Valor del tipo de registro de Registro en Clientes	Mostrar
10	VAL00000000		Valor del tipo de registro de Registro en Departamentos	Mostrar

Imagen 6: Opciones del Usuario

Para leer el valor actual de una opción global se usa la siguiente sintaxis:

```
{{goOpcionGlobal.OPCION}}
```

Donde OPCION es el código de la opción que se desea leer. Por ejemplo, para determinar si el manejo de lotes está activo en la empresa actual, en el sistema administrativo, se usaría:

```
{{goOpcionGlobal.ACTLOT}}
```

Si la opción global solicitada existe entonces el evento no seguirá ejecutándose.

De forma similar, para leer el valor actual de una opción de usuario se usa la siguiente sintaxis:

```
{{goOpcionUsuario.OPCION}}
```

Donde OPCION es el código de la opción que se desea leer. Si el usuario que esté activo al momento de ejecutarse el evento no tiene creada y activa la opción indicada, entonces le leerá la opción con el mismo código entre las opciones globales. Si la opción global tampoco existe entonces el evento no seguirá ejecutándose.

Objeto de Datos de Ejecución

Después de ejecutarse el código del campo "Datos de Ejecución" (descrito más adelante en la sección "Datos y Condición de Ejecución") se tiene disponible el objeto lvData, que contiene el resultado de una o más sentencias SELECT ejecutadas en dicho campo.

Para leer el contenido de este objeto se usa la sintaxis siguiente:

```
{{lvData[M][N].CAMPO}}
```

Donde "M" es el número de la tabla a leer (la primera tabla será 0, la segunda 1, etc.). "N" es el número de renglón a leer (comenzando por 0 para el primer renglón). Y "CAMPO" es el nombre de un campo del renglón indicado por "M" y "N".

Datos Básicos del Evento

Se recomienda que al crear un evento se le dé un nombre claro que índice en líneas generales la tarea realizada por el mismo y el formulario y gancho sobre el cual se ejecutará. Algunos nombres recomendados son:

- Validación de Descuento Global en Facturas de Venta antes de Guardar Nuevo.
- Validación de Descuento Global en Facturas de Venta antes de Guardar Editado.
- Validación de Estado de Crédito antes de Confirmar Pedidos.
- Validación de Estatus de Factura de Ventas Antes de Imprimir el Formato.
- Validación de Monto de Factura y Envío de Correo Antes de Guardar Nuevo.
- Validación de Nivel del Usuario Antes de Eliminar Cobro a Clientes.

Cómo puede observarse, los eventos “Antes de guardar...” se programan en pares, y normalmente serán dos eventos iguales. Más detalle acerca de los ganchos en la sección “Información del Gancho y Formulario Asociado”.

Igualmente, debe asignársele un código que resuma el nombre del mismo. Por ejemplo “VALDESFV1”, “VALDESFV2” y “VALCRCOPED” para los primeros tres eventos de la lista anterior.



Imagen 7: Administrador de Eventos – Sección Datos

En esta misma sección se debe indicar el tipo de disparador del evento.

Actualmente, solo está disponible el disparador de tipo “Evento”. Los disparadores por Alertas o Tiempo estarán disponibles a partir de la versión 3.13.9 de eFactory.

La fecha y hora de inicio solo aplica para el disparador de tipo “Alerta”.

Los campos Comentario y Notas servirán para dar una explicación más detallada del funcionamiento del evento.

Datos y Condición de Ejecución

Parte de la programación de los eventos hace uso del lenguaje SQL, aunque la mayor parte de la programación se realizará mediante LIF (Lenguaje Interpretado Factory). Este ofrece una sintaxis simplificada de expresiones, además del acceso a objetos y propiedades comunes de la empresa o usuario actual.

El campo **Datos de Ejecución** (Imagen 8) contiene una expresión SQL que se usará preparar los datos que serán usados durante la ejecución del evento.



Imagen 8: Administrador de Eventos – Detalle de la Ejecución

Este campo consistirá típicamente de código SQL arbitrario que terminará con al menos un SELECT final que devolverá la tabla de datos usada por el evento (el objeto `lvData`). El código SQL utilizado se ejecutará sobre la base de datos de la empresa, por lo que tendrá acceso a cualquier tabla, función, vista o procedimiento almacenado en ella. Adicionalmente, incluirá algunas referencias a la vista y a otros objetos disponibles indicados en la sección "El Lenguaje Interpretado Factory".

Por ejemplo: Si deseamos programar el evento "Validación de Descuento Global en Facturas de Venta antes de Guardar Nuevo" (asociado al gancho "Antes de Guardar Nuevo" del formulario de Facturas de Venta), el campo podría tener el código SQL siguiente:

```
SELECT Cod_Cli, Nom_Cli, Por_Des, Tip_Cli
FROM Clientes
WHERE Cod_Cli = {{loVista.Cod_Cli}}
```

Donde la expresión `{{loVista.Cod_Cli}}` hace referencia al campo `Cod_Cli` (Código del Cliente) del encabezado de la factura que está siendo guardada.

El campo **Ejecutar Cuando** contiene una expresión LIF que indicará bajo qué condición se ejecutará el resto del evento. En algunos casos queremos que el resto de evento se ejecute siempre, por lo que guardaremos el valor TRUE (Imagen 8).

En otros casos queremos que el resto del evento (el envío de correo, los avisos...) solo se ejecute si se cumple cierta condición. Por ejemplo, si se coloca la siguiente expresión en el campo "Ejecutar Cuando":

```
{{lvData[0][0].Tip_Cli}} == "NORMAL" OR ({{loVista.Mon_Net}}*{{loVista.Tasa}} <= 1000)
```

El resto del evento solo se ejecutaría si el Tipo de Cliente seleccionado es igual al texto "Normal" o si el monto neto del documento es menor a 1.000 (monto en moneda base).

Envío de Correo

Si la expresión almacenada en el campo "Ejecutar Cuando" devuelve verdadero (TRUE) entonces se enviará un correo con el contenido indicado en la sección "Enviar Correo".

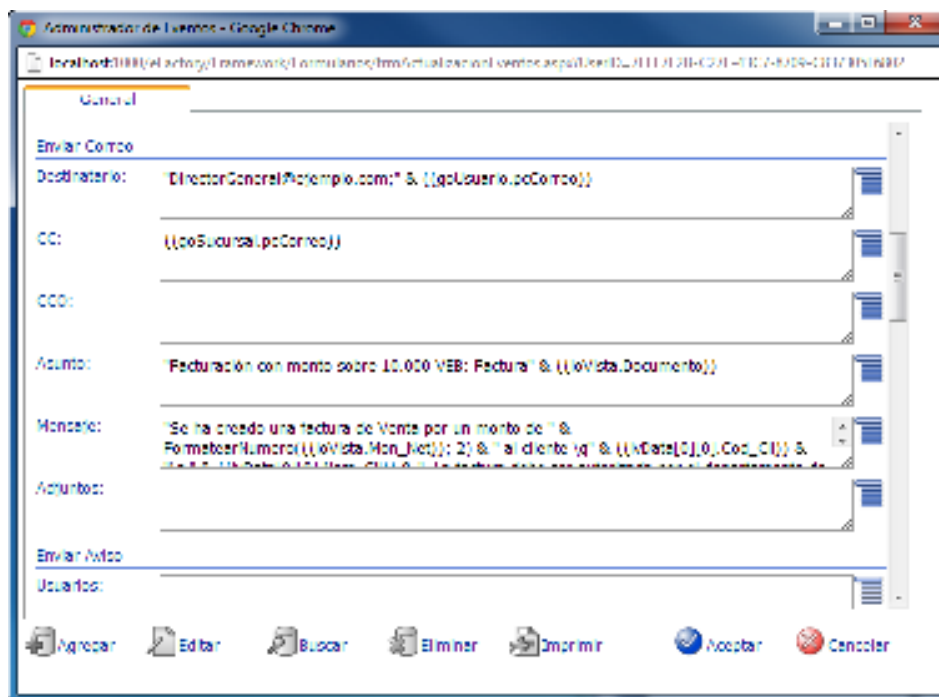


Imagen 9: Administrador de Eventos – Enviar Correo

Los campos "Destinatario", "CC" u "CCO" de esta sección pueden recibir una expresión LIF que devuelva una cadena con uno o más destinatarios, separados por punto y coma (;). Debe haber al menos una dirección de correo válida en el campo "destinatario" para que el correo sea enviado.

Los campos "Asunto" y "Mensaje" permiten especificar la información que será enviada en el correo. Por ejemplo, si se indica el siguiente código LIF en el campo "Mensaje":

```
"Se ha creado una Factura de Venta por un monto de " & FormatearNumero({{loVista.Mon_Net}}; 2) &
({{loVista.Cod_Mon}} & " al cliente \" & {{lvData[0][0].Cod_Cli}} & "\" & {{lvData[0][0].Nom_Cli}} &
". La factura debe ser autorizada por el departamento de cobranzas antes de ser despachada."
```

El contenido del correo generado será similar al siguiente:

```
Se ha creado una Factura de Ventas por un monto de 14.128,17 VEB al cliente 'C0007108' Centro de
Inversiones Empresariales. La factura debe ser autorizada por el departamento de cobranzas antes de ser
despachada.
```

La opción de envío de archivos adjuntos estará disponible a partir de la versión 3.13.9 de eFactory.

Envío de Avisos

Si la expresión almacenada en el campo "Ejecutar Cuando" devuelve verdadero (TRUE) entonces se enviará un mensaje a los usuarios indicados mediante el contenido de la sección "Enviar Aviso".

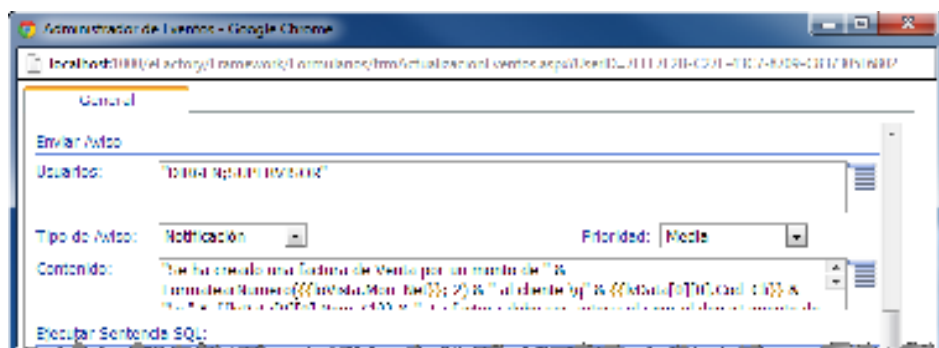


Imagen 10: Administrador de Eventos – Enviar Aviso

Los campos Tipo de Aviso y Prioridad no tienen incidencia en el comportamiento del evento, sino que son utilizados para clasificar los avisos generados.

El campo "Usuarios" permite especificar, mediante una expresión LIF, una lista de códigos de usuario separados por punto y coma, a los cuales se les enviará un mensaje por medio del sistema de mensajes en pantalla de eFactory. Igualmente, el contenido del mensaje se indica mediante una expresión LIF el campo "Contenido".



Imagen 11: Aviso Generado Desde un Evento

Los avisos se mostrarán en el Panel de Mensajes y Avisos en la ventana principal de eFactory (Imagen 11).

Una versión más interactiva del panel de Mensajes y Avisos estará disponible a partir de la versión 3.13.9 de eFactory.

Ejecución SQL Arbitraria

Si la expresión almacenada en el campo "Ejecutar Cuando" devuelve verdadero (TRUE) entonces el bloque de código SQL de esta sección será ejecutado.

Esta sección consiste en un único campo que permite ejecutar código SQL de propósito general, por ejemplo para actualizar un registro de la base de datos o ejecutar un procedimiento almacenado existente.

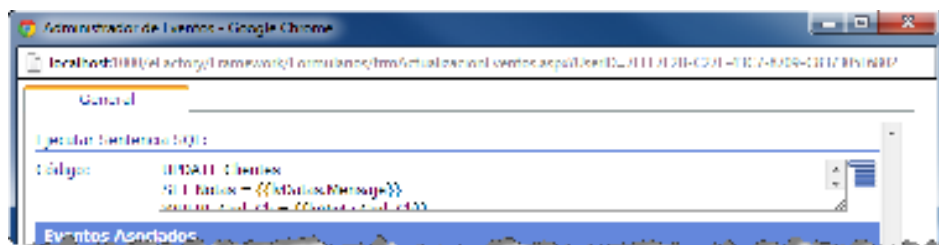


Imagen 12: Administrador de Eventos – Ejecutar Sentencia SQL

En este campo se puede hacer referencia a cualquiera de los objetos definidos anteriormente. Por ejemplo, la siguiente expresión actualizará el campo Notas del cliente correspondiente al documento actual.

```
UPDATE Clientes
SET Notas = {{lvData.Mensaje}}
WHERE Cod_Cli = {{lvData.Cod_Cli}}
```

Se recomienda que cualquier actualización importante ejecutada dentro esta sección incluya el guardado de las auditorías correspondientes. Para esto se puede hacer uso del Procedimiento Almacenado `SP_GuardarAuditoria`:

```
EXECUTE @RC = [dbo].[sp_GuardarAuditoria]
    @lcUsuario, @lcTipo, @lcTabla, @lcOpcion, @lcAccion,
    @lcDocumento, @lcCodigo, @lcClave2, @lcClave3, @lcClave4, @lcClave5,
    @lcDetalle, @lcEquipo, @lcSucursal, @lcObjeto, @lcNotas, @lcEmpresa
```

Información del Gancho y Formulario Asociado

El contenido de esta sección solo aplica para los eventos con disparador de tipo "Evento".

En la sección de Eventos Asociados se indica a qué "elemento" de eFactory estará asociado el evento y cuál será el gancho que lo ejecutará.

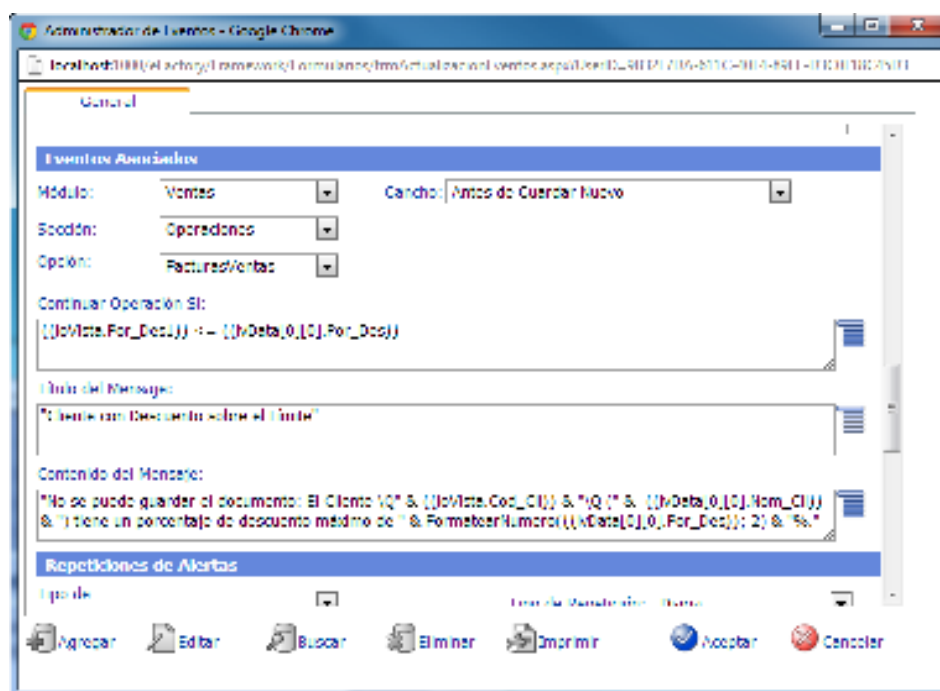


Imagen 13: Administrador de Eventos - Eventos Asociados

Por medio de las listas desplegables de Módulo, Sección y Opción se puede seleccionar el formulario al cual se asociará el evento. La lista desplegable Gancho muestra todos los ganchos disponibles en el sistema, sin embargo no todos los ganchos están disponibles en todos los formularios.

Por ejemplo, los ganchos relativos a Agregar, Editar, Eliminar, Guardar y Cancelar están disponibles en todas las actualizaciones (Clientes, Artículos, Trabajadores, Cuentas Contables...) y Documentos (Facturas Cotizaciones, Recibos de Pago, Comprobantes...), pero los ganchos relativos a Confirmar, Reversar y Anular solo están disponibles en Documentos que incluyan esa acción.

El contenido de los siguientes tres campos: "Continuar Operación Si", "Título del Mensaje" y "Contenido del Mensaje" solo aplica para los ganchos de tipo "Antes de..." y su intención es proveer un mecanismo que permita bloquear la acción del usuario antes de que esta se ejecute si no cumple con la condición especificada.

El campo "**Continuar Operación Si**" contiene una expresión LIF que devuelve un valor lógico (Verdadero o Falso) que indica si la operación ejecutada por el usuario está permitida o debe ser bloqueada. Si la operación será siempre permitida (por ejemplo, si la única tarea del evento es enviar una notificación) entonces se puede especificar el valor fijo "TRUE".

Los campos "**Título del Mensaje**" y "**Contenido del Mensaje**" contienen cada uno una expresión LIF que devuelve una cadena de texto, y permiten establecer la información que se mostrará en pantalla en un mensaje modal para indicar al usuario la razón por la cual no se pudo completar la acción solicitada (no debe confundirse con lo indicado en la sección "Envío de Avisos"). Este mensaje solo se mostrará cuando la acción sea bloqueada por el campo "Continuar Operación Si". En caso de que la operación esté siempre permitida, se puede especificar una cadena vacía (con dos comillas dobles: "") en ambos campos.

Repeticiones de Alertas de Tiempo

El contenido de esta sección solo aplica para los eventos con disparador de tipo "Alerta".

Las repeticiones de Alertas de Tiempo permitirán programar eventos para que se ejecuten en un momento determinado o en intervalos de tiempo regulares: a diario, cada domingo, el segundo lunes de cada mes, cada 1º de Mayo, etc.

Imagen 14: Repeticiones por Alertas de Tiempo

Estos podrían ejecutar tareas de mantenimiento, copias de seguridad, actualización de estadísticas, y cualquier otra que no dependan directamente del usuario.

Como se mencionó en una sección anterior, los disparadores por Alertas estarán disponibles a partir de la versión 3.13.9 de eFactory.

Ejemplos de Eventos

Los siguientes son ejemplos completos de eventos configurados en eFactory.

Ejemplo #1: Validación de descuentos antes de guardar

Requerimiento: Se solicita impedir que un usuario con nivel menor a 100 pueda crear una Factura de Ventas con un descuento global superior al descuento asignado al cliente.

Consideraciones: Para garantizar que por ningún medio se aplique un descuento mayor al indicado al cliente es necesario:

- Bloquear la edición de precios y tipo de precio (Imagen 15) por medio de las opciones: "BLOPREFAC: Bloquear Columna de Precio en Facturas" y "BSELPREFAC: Bloquear la Selección del Tipo de Precio (1-5) en Facturas". Así solo se podrá aplicar el precio original del artículo.
- Desbloquear la edición de descuentos por medio de la opción "BDESMANFAC: Bloquear Descuentos Manuales en Facturas". Esto, debido a que queremos que el usuario especifique manualmente un descuento global si se requiere.
- Bloquear los Campos correspondientes a Otros Montos, ya que estos permiten ingresar valores negativos que disminuirán el monto de la factura (actuando como un descuento).

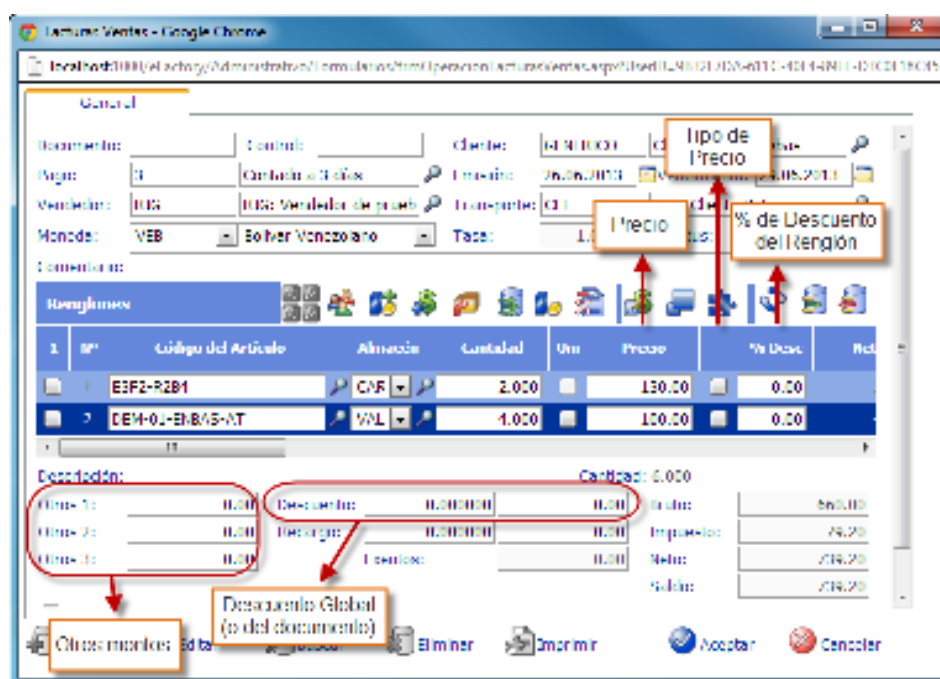


Imagen 15: Ejemplo #1 – Elementos de Facturas de Venta a considerar

Procedimiento: Crearemos un nuevo Evento con los siguientes Valores:

- El código y nombre del nuevo evento será "VALDESFV1" y "Validación de Descuento Global en Facturas de Venta antes de Guardar Nuevo" respectivamente.
- Le indicaremos un estatus Activo y el disparador será de tipo "Evento". También agregamos un comentario descriptivo: "Verifica que el porcentaje de ganancia de la factura de ventas no sea mayor al porcentaje de descuento global asignado al cliente. La validación no se ejecuta si el usuario tiene nivel 100."
- En Datos de Ejecución usaremos la siguiente expresión SQL (con referencias al código del cliente, tomado desde loVista):

```

SELECT Cod_Cli, Nom_CLI, Por_Des
FROM Clientes
WHERE Cod_Cli = {{loVista.Cod_CLI}}

```

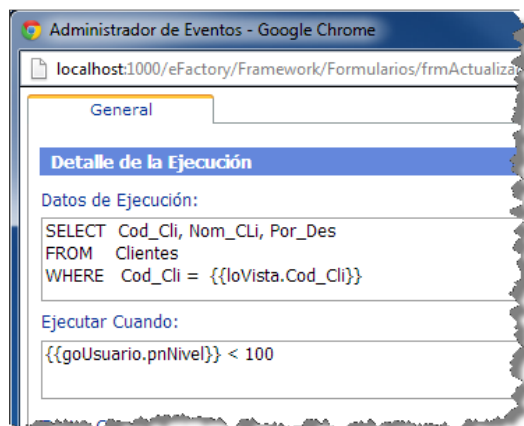


Imagen 16: Ejemplo #1 – Detalles de la ejecución

- En Ejecutar Cuando usaremos la siguiente expresión LIF para indicar que el evento solo se ejecutará si el nivel del usuario actual (el que esté guardando la factura de ventas) es menor a 100 (el nivel se especifica en el formulario de Usuarios Globales, en el módulo de sistemas, Imagen 17):

```
{{goUsuario.pnNivel}} < 100
```

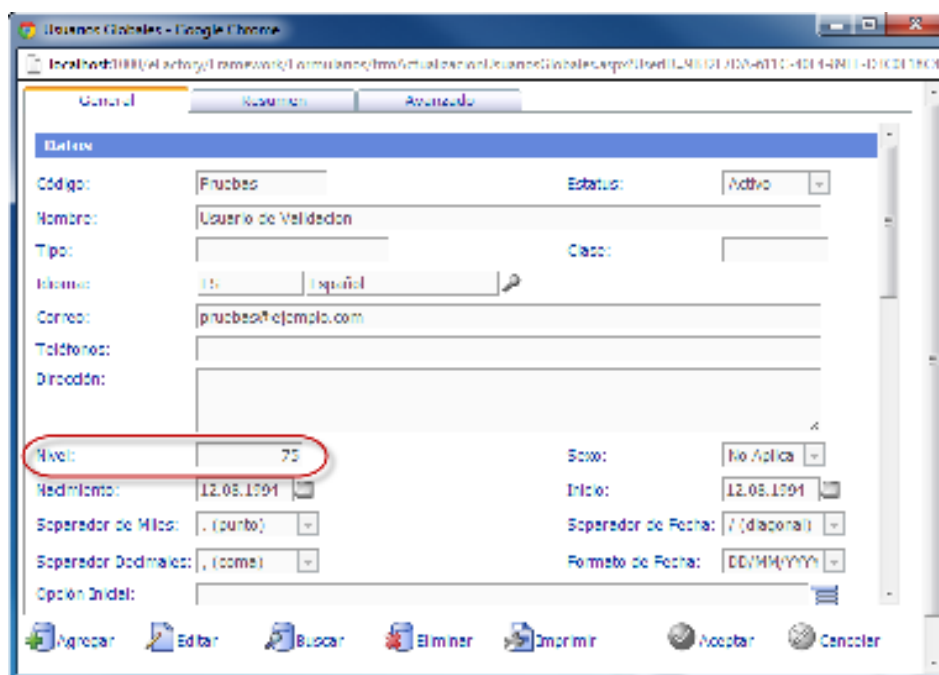


Imagen 17: Nivel del Usuario Global

- En la sección de Eventos Asociados (Imagen 13) indicaremos que este evento se asocia al Módulo "Ventas", Sección "Operaciones", Opción "FacturasVentas"; e indicaremos que el gancho disparador será "Antes de Guardar Nuevo". Adicionalmente, los siguientes tres campos tendrán las expresiones LIF siguientes:

Continuar Operación Si:

```
{{loVista.Por_Des1}} <= {{lvData[0][0].Por_Des}}
```

Título del Mensaje:

"Cliente con Descuento sobre el Límite"

Contenido del Mensaje:

"No se puede guardar el documento: El Cliente \Q" & {{loVista.Cod_Cli}} & "\Q (" & {{lvData[0][0].Nom_Cli}} & ") tiene un porcentaje de descuento máximo de " & FormatearNumero({{lvData[0][0].Por_Des}}; 2) & "%."

- Guardamos el Evento y lo duplicamos (ficha Avanzados, botón Duplicar). En el nuevo evento solo cambiaremos el código a "VALDESFVG2", el nombre será "Validación de Descuento Global en Facturas de Venta antes de Guardar Editado" y el gancho asociado será "Antes de Guardar Editado".

Ahora, cuando un usuario con nivel inferior a 100 intente guardar una factura de Ventas nueva (o guardar cambios en una existente) se verificará que no se haya aplicado un documento de documento superior al especificado para el cliente. En caso de que se indique en el documento un descuento mayor se verá un mensaje como el siguiente y no permitirá guardar el documento:

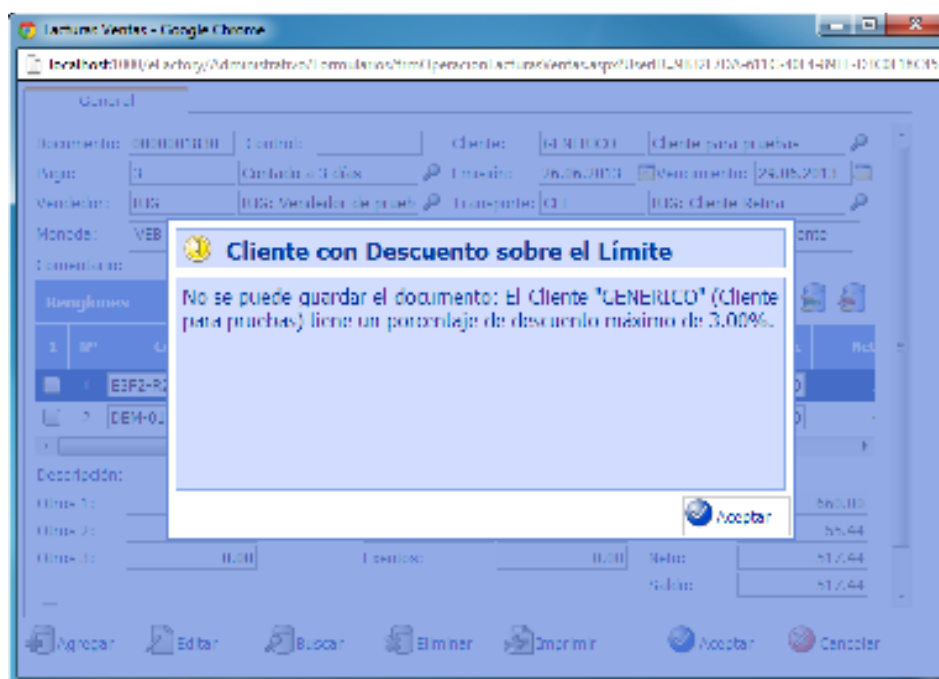


Imagen 18: Mensaje Generado Desde un Evento

Consideraciones Adicionales: Para este ejemplo básico no se tomaron en cuenta los descuentos de los renglones (los cuales no se considerarían en la validación anterior). Alternativamente, en lugar de bloquear los Otros Montos, podrían mantenerse desbloqueados y se validaría dentro del evento que el usuario no haya ingresado un monto negativo en ninguno de ellos.

Ejemplo #2: Validación de Ganancia Antes de Confirmar

Requerimiento: Se solicita impedir que un usuario pueda confirmar una Cotización si al menos uno de los artículos cotizados genera una ganancia inferior a lo configurado para el artículo.

Consideraciones: En este caso, la ganancia no se verificará al guardar el documento, sino al confirmarlo. Con esto se da flexibilidad para cargar las compras de estos artículos y recalcular los costos después de haber creado la cotización (mediante el proceso de recosteo, que reasigna los costos de los artículos en los documentos correspondientes): Así, podemos asegurar que los costos usados para la validación de la ganancia estén actualizados.

Procedimiento:

- Crearemos un nuevo evento, y le asignamos los datos básicos (código, nombre, tipo, comentario...) de la misma forma que lo hicimos en el ejemplo #1.
- La mayor parte de la programación se realizará en el campo Datos de Ejecución, mediante el siguiente código SQL:

```
--Parámetros globales
DECLARE @lnTasaOtraMoneda DECIMAL(28,10);
SET @lnTasaOtraMoneda = {{goEmpresa.pnTasaMonedaAdicional}};

DECLARE @llGananciaSobreCosto BIT;
SET @llGananciaSobreCosto = {{goOpcionGlobal.GANCOSPRE}};

--Tabla de datos
CREATE TABLE #tmpRenglones(
    Renglon INT,
    Cod_Art VARCHAR(30),
    Precio DECIMAL(28,10),
    Por_Gan DECIMAL(28,10),
    Tip_Pre CHAR(1),
    Nom_Art VARCHAR(100),
    Costo DECIMAL(28,10)
)

INSERT INTO #tmpRenglones
SELECT RCot.Renglon AS Renglon,
       RCot.Tip_Pre AS Tip_Pre,
       RTRIM(RCot.Cod_Art) AS Cod_Art,
       RTRIM(Art.Nom_Art) AS Nom_Art,
       RCot.Precio1* RCot.Tasa AS Precio,
       (CASE Art.Cos_Pre
        WHEN 'Promedio MP' THEN RCot.Cos_Prol
        WHEN 'Ultimo MP' THEN RCot.Cos_Ult1
        WHEN 'Anterior MP' THEN RCot.Cos_Ant1
        WHEN 'Promedio MS' THEN RCot.Cos_Pro2*@lnTasaOtraMoneda
        WHEN 'Ultimo MS' THEN RCot.Cos_Ult2*@lnTasaOtraMoneda
        WHEN 'Anterior MS' THEN RCot.Cos_Ant2*@lnTasaOtraMoneda
        ELSE RCot.Cos_Ult1
       ) AS Costo,
       Art.Gan_Min AS Por_Gan
FROM Renglones_Cotizaciones AS RCot
JOIN Cotizaciones AS Cot ON Cot.Documento = RCot.Documento
JOIN Articulos AS Art ON Art.Cod_Art = RCot.Cod_Art
WHERE Cot.Documento = {{loVista.Documento}}

--Cálculo de Ganancia
DECLARE @llRenglonesValidos BIT
DECLARE @lnRenglon INT
DECLARE @lnTotal INT
DECLARE @lcMensajeRenglon VARCHAR(MAX)

SET @llRenglonesValidos = 1;
SET @lnRenglon = 1;
SET @lnTotal = (SELECT COUNT(*) FROM #tmpRenglones);
SET @lcMensajeRenglon = '';

DECLARE @lnGanRenglon DECIMAL(28,10);
DECLARE @lnGanArticulo DECIMAL(28,10);
DECLARE @lcArticulo VARCHAR(140);

WHILE(@lnRenglon <= @lnTotal)
BEGIN
```

```

SELECT      @lnGanRenglon = (CASE WHEN (@llGananciaSobreCosto=1)
                                THEN (CASE WHEN Costo>0 THEN (Precio - Costo)*100/Costo ELSE 0 END)
                                ELSE (Precio - Costo)*100/Precio
                                END),
            @lnGanArticulo = Por_Gan,
            @lcArticulo    = Cod_Art + ' (' + Nom_Art + ' )'
FROM        #tmpRenglones
WHERE       Renglon = @lnRenglon

IF( @lnGanRenglon<@lnGanArticulo)
BEGIN
    SET @lcMensajeRenglon = @lcMensajeRenglon + '* Renglón ' + CAST(@lnRenglon AS VARCHAR(3)) +
    ': Artículo ' + @lcArticulo + CHAR(13);
    SET @llRenglonesValidos = 0;
END
SET @lnRenglon = @lnRenglon + 1
END

DROP TABLE #tmpRenglones

-- TABLA ÚNICA DE DATOS
SELECT      @llRenglonesValidos AS RenglonesValidos,
            @lcMensajeRenglon  AS MensajeRenglon

```

En el código anterior, se comienza por obtener la tasa de la moneda secundaria de la empresa (para los costos en moneda adicional), y la opción global "GANCOSPRE" (que indica cual será la base para el cálculo de ganancia).

Luego se guarda en una tabla temporal los datos de los artículos, sus precios (tomados del documento y convertidos a la moneda base mediante la tasa del mismo), sus costos (tomados del documento y convertidos a la moneda base mediante la tasa de "la otra moneda") y el porcentaje de ganancia mínima configurado (tomado de la tabla de artículos).

El siguiente bloque de código recorre los renglones de la tabla temporal (mediante un bucle WHILE) y calcula la ganancia real del renglón, y si esta es menor a la mínima ganancia del artículo activa la bandera correspondiente y prepara el mensaje a mostrar.

Finalmente se ejecuta un SELECT que contendrá los datos que podrán usarse en el resto del evento, mediante el objeto lvData. Es importante destacar que, aun cuando este campo contiene mucho código SQL, el resultado de éste consiste en una sola tabla con un solo renglón, que será usado en el resto del evento: este es el patrón de trabajo recomendado al programar eventos.

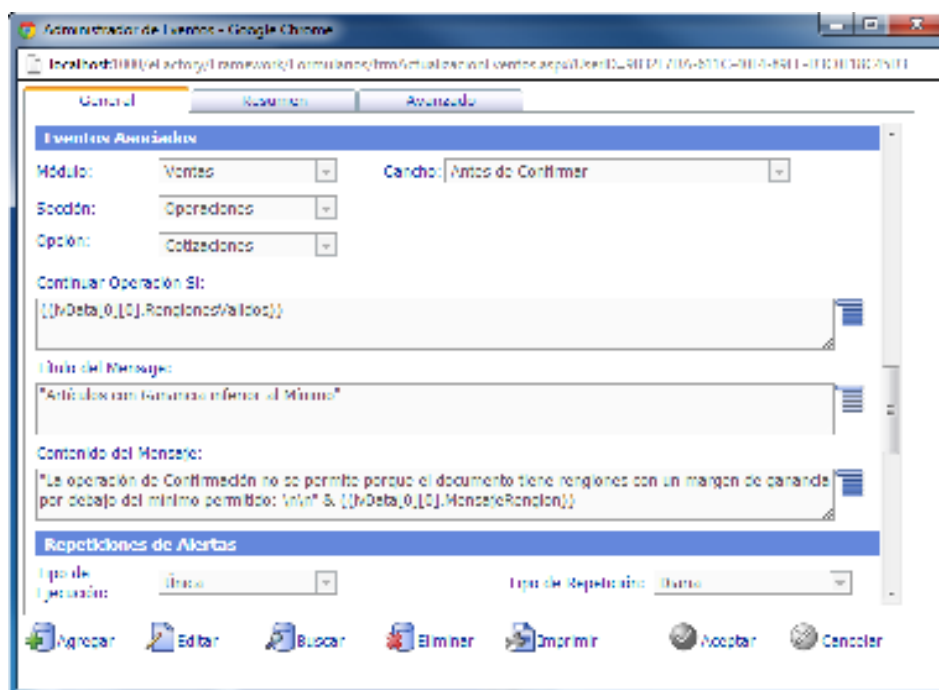


Imagen 19: Ejemplo #2 - Eventos Asociados y Mensaje

- En el campo "Ejecutar Cuando" dejaremos el valor TRUE, ya que para todas las cotizaciones se aplicará la validación.
- El evento será asociado al Módulo de "Ventas", Sección "Operaciones", Opción "Cotizaciones", en el gancho "Antes de Confirmar".
- En el campo "Continuar Operación Si" asignaremos la siguiente expresión, que según lo indicado en el código del campo "Datos de Ejecución" nos indicará si los renglones son válidos (ganancia mayor o igual a la especificada) o no:

```
{{lvData[0][0].RenglonesValidos}}
```

- Finalmente, se prepara el mensaje al usuario en caso de que los renglones no sean válidos:

Título del Mensaje:

"Artículos con Ganancia Inferior al Mínimo"

Contenido del Mensaje:

"La operación de Confirmación no se permite porque el documento tiene renglones con un margen de ganancia por debajo del mínimo permitido: \n\n" & {{lvData[0][0].MensajeRenglon}}

Nótese que el valor {{lvData[0][0].MensajeRenglon}} ya contiene todo el detalle de los renglones que no cumplen con la ganancia establecida.

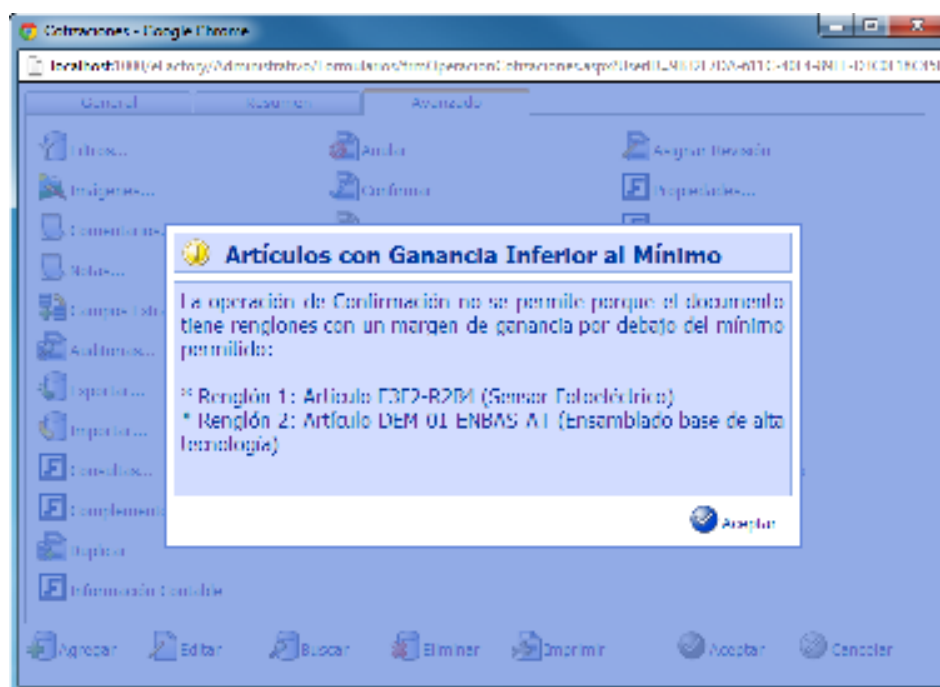


Imagen 20: Ejemplo #2 – Mensaje al no Permitir Confirmar

Otros Ejemplos de Aplicación

- Es posible validar si el documento está confirmado antes de imprimir el formato correspondiente. También es posible consultar las Propiedades y Campos Extra del documento para impedir la impresión en caso de que el documento no esté debidamente autorizado.
- Se puede verificar antes de guardar el documento, que el vendedor asignado al mismo cumpla con alguna condición relacionada a la gestión de la fuerza de ventas, por ejemplo, que el vendedor no esté temporalmente suspendido, que tenga permiso para vender al cliente seleccionado, que solo tenga permitido emitir documentos de contado, que no pueda vender ciertos artículos, o artículos de cierto departamento, etc.
- Se puede verificar al crear un nuevo cliente que se haya ingresado información que para el proceso de negocio sea importante, por ejemplo una dirección de correo, un contacto, el teléfono o el sitio web de la compañía.
- Antes de crear una factura de compras se puede verificar que ésta provenga de una Orden de Compra o de una Requisición y que ésta esté debidamente autorizada por el departamento de Compras. Adicionalmente se puede enviar una notificación automática o un correo a un supervisor o directivo cuando un empleado nuevo cargue una compra por un monto superior a cierto límite.
- Se puede programar un procedimiento almacenado y vincularlo a un evento para que se ejecute en un momento dado. Por ejemplo: se puede actualizar los márgenes de ganancia o los precios de un artículo al confirmar una factura de compras.