

Fábrica - Modelos Logs - API de Mensajes

Esta sección describe el API de mensajes que debe ser utilizada en las aplicaciones de la compañía.

Tipos de Mensajes

A continuación se listan los tipos de mensajes que deben ser considerados en el log.

- Mensajes Informativos: Utilizados para informar al cliente sobre procesos, estados, opciones, confirmaciones etc.
- Mensajes de Advertencias: Utilizados para alertar sobre situaciones en las aplicaciones.
- Mensajes de Errores: Utilizados para notificar errores generados en los procesos de la aplicaciones.

Que no se considera en el API

- Textos descriptivos de procesos como (guías, tutoriales).
- Mensajes asociados a sistemas operativos, maquinas virtuales, bases de datos o errores desconocidos.

¿Donde usar el Servicio?

Este servicio debe consumirse en las generaciones de errores de:

- Aplicaciones Java
- Aplicaciones .Net
- Web Service
- Integración Supertabla
- Soluciones que afecten los productos SICOF

Operaciones del API

El API soporta las siguientes operaciones:

- Consultar mensaje por código interno numérico. Columna: **CODIGO_MENSAJE**
- Consultar mensaje por código externo texto. Columna **CODIGO**
- Consultar mensaje formateado por código interno numérico y array de (String) literales # (Ejemplo: Mensaje = Hola #1 | Literal #1 = Mundo | Mensaje Formateado = Hola Mundo)
- Consultar mensaje formateado por código externo texto y array de (String) literales # (Ejemplo: Mensaje = Hola #1 | Literal #1 = Mundo | Mensaje Formateado = Hola Mundo)

Nota: Aplicaciones Powerbuilder

Powerbuilder tiene restricciones para el consumo de servicios Rest por lo tanto en las aplicaciones de esta tecnología se implementará un [API](#) para realizar las llamadas.

Diccionario de Datos

OWNER	PRESUP01	TABLE	MENSAJES_SISTEMA	COMMENTS	Tabla que contiene los mensajes de los procesos de SICO
#	NAME	NULLEABLE	TYPE	COMMENTS	
1	CODIGO_MENSAJE	N	NUMBER	Código Interno del Mensaje (Es unico en la tabla y se asigna de forma manual)	
2	MENSAJE	Y	VARCHAR2(1024)	Descripción del mensaje que se representa puede registrar expresiones de sustitución con la regla #identificador numérico Ej Hola #1	
3	INFORMATION	Y	CHAR(1)	Bandera que puede ser utilizada por el frontend: indica que el mensaje es de tipo: Información	
4	STOP	Y	CHAR(1)	Bandera que puede ser utilizada por el frontend: indica que el mensaje es de tipo: Error	
5	EXCLAMATION	Y	CHAR(1)	Bandera que puede ser utilizada por el frontend: indica que el mensaje es de tipo: Advertencia	
6	QUESTION	Y	CHAR(1)	Bandera que puede ser utilizada por el frontend: indica que el mensaje es de tipo: Pregunta o Interrogación	
7	NONE	Y	CHAR(1)	Bandera que puede ser utilizada por el frontend: indica que el mensaje no especifica su tipo: Por lo general se usa Información	
8	OK	Y	CHAR(1)	Bandera que puede ser utilizada por el frontend: indica que el mensaje utiliza boton: Aceptar	
9	CANCEL	Y	CHAR(1)	Bandera que puede ser utilizada por el frontend: indica que el mensaje utiliza boton: Cancelar	
10	YES	Y	CHAR(1)	Bandera que puede ser utilizada por el frontend: indica que el mensaje utiliza boton: Si	

11	NO	Y	CHAR(1)	Bandera que puede ser utilizada por el frontend: indica que el mensaje utiliza boton: No
12	RETRY	Y	CHAR(1)	Bandera que puede ser utilizada por el frontend: indica que el mensaje utiliza boton: Reintentar
13	ABORT	Y	CHAR(1)	Bandera que puede ser utilizada por el frontend: indica que el mensaje utiliza boton: Abortar
14	IGNORE	Y	CHAR(1)	Bandera que puede ser utilizada por el frontend: indica que el mensaje utiliza boton: Ignorar
15	DEFAULTBUTTON	Y	NUMBER(1,0)	Bandera que puede ser utilizada por el frontend: indica que el mensaje utiliza un boton por defecto (con foco) es un identificador numérico que se representa de izuiquerda a derecha
16	CODIGO_BASE_DATOS	Y	NUMBER(10,0)	Código interno de base de datos que está asociado al mensaje
17	USUARIO_EMPRESA	Y	VARCHAR2(30)	Usuario Empresa
18	CODIGO_MEMPRESA	Y	VARCHAR2(50)	Código de la emprsra por defecto '9999999999' para entorno uniempresa.
19	FECHA_REGISTRO	Y	DATE	Fecha del sistema
20	TIPO_PROCESO	Y	VARCHAR2(256)	Clasificación del mensaje Ej: Disponiiblidad, Compromiso etc.
21	CODIGO	Y	VARCHAR2(128)	Es un identificador string unico que puede ser utilizado para identificar el mensaje por lo general inicial con una sigla nemotecnica asociada a la clasificación (TIPO_PROCESO) EJ: Para Disponibilidad seria CDP00001
22	TITULO	Y	VARCHAR2(256)	Campo que puede ser utilizada por el frontend para visualizar un titulo en el mensaje.
23	OBSERVACIONES	Y	VARCHAR2(4000)	Información complementaria que orienta al usuario sobre la situación notificada en el mensaje. Puede llevar pasos, consejos o indicaciones.

Proceso de Creación de Mensajes

Todos los mensajes utilizados en las aplicaciones deben ser registrados en esta tabla. A continuación se describen los campos que deben ser incluidos en la creación segun el tipo de mensaje:

#	Campo	Observación
1	CODIGO_MENSAJE	Requerido: Se debe enviar el siguiente numero disponible en la columna. Puede utilizar la siguiente sentencia para asegurar la condición requerida: SELECT NVL(MAX(CODIGO_MENSAJE), 0) + 1 FROM PRESUP01.MENSAJES_SISTEMA
2	MENSAJE	Requerido: El mensaje debe claro y conciso. No debe generar ambigüedades en los usuarios y no debe estar expresado en lenguaje técnico.
3	INFORMATION	Esta columna debe ser enviada con el valor 'S' si el mensaje es Informativo . De lo contrario debe ser enviado en 'N'
4	STOP	Esta columna debe ser enviada con el valor 'S' si el mensaje es de Error . De lo contrario debe ser enviado en 'N'
5	EXCLAMATION	Esta columna debe ser enviada con el valor 'S' si el mensaje es de Advertencia . De lo contrario debe ser enviado en 'N'
6	QUESTION	Esta columna debe ser enviada con el valor 'S' si el mensaje es de Confirmación . De lo contrario debe ser enviado en 'N'
7	NONE	Esta columna debe ser enviada con el valor 'S' si el mensaje no lleva ícono. De lo contrario debe ser enviado en 'N'
8	OK	Esta columna debe ser enviada con el valor 'S' cuando el mensaje visualiza boton Aceptar . De lo contrario debe ser enviado en 'N'
9	CANCEL	Esta columna debe ser enviada con el valor 'S' cuando el mensaje visualiza boton Cancelar . De lo contrario debe ser enviado en 'N'
10	YES	Esta columna debe ser enviada con el valor 'S' cuando el mensaje visualiza boton Si . De lo contrario debe ser enviado en 'N'
11	NO	Esta columna debe ser enviada con el valor 'S' cuando el mensaje visualiza boton No . De lo contrario debe ser enviado en 'N'
12	RETRY	Esta columna debe ser enviada con el valor 'S' cuando el mensaje visualiza boton Reintentar . De lo contrario debe ser enviado en 'N'
13	ABORT	Esta columna debe ser enviada con el valor 'S' cuando el mensaje visualiza boton Abortar . De lo contrario debe ser enviado en 'N'
14	IGNORE	Esta columna debe ser enviada con el valor 'S' cuando el mensaje visualiza boton Ignorar . De lo contrario debe ser enviado en 'N'
15	DEFAULTBUTTON	Esta columna debe ser enviada con uno de los siguientes valores [1, 2, 3] cuando el mensaje visualiza el foco en uno de los botones utilizados. De lo contrario se ignora esta columna.
16	CODIGO_BASE_DATOS	Solo se utiliza cuando el mensaje está asociado a un Código de error de base de datos. De lo contrario se ignora esta columna.
17	USUARIO_EMPRESA	Enviar 'PRESUP01'
18	CODIGO_MEMPRESA	Enviar '9999999999'
19	FECHA_REGISTRO	Enviar SYSDATE
20	TIPO_PROCESO	Requerido: Clasificación del mensaje Ej: Disponiibilidad, Compromiso etc.
21	CODIGO	Requerido: Código Texto del Mensaje Ej: TERCERO_NOT_FOUND, CDP_NO_BALANCE etc.
22	TITULO	Requerido: Título del mensaje el cuál es utilizado en la barra de titulo de la ventana.
23	OBSERVACIONES	Este campo solo es requerido cuando el mensaje requiere gestión y debe presentar información complementaria que oriente al usuario sobre la situación notificada en el mensaje.

Reglas para la columna: OBSERVACIONES

Si un mensaje debe mostrar más información respecto a la situación notificada se debe utilizar esta columna. Sin embargo debe tener presente la siguientes indicaciones:

- Si el mensaje requiere gestión de un proceso esta columna se debe describir con una serie de pasos de las acciones a realizar.
- Si el mensaje indica una situación puntual esta columna debe indicar la razón de la situación presentada e indicar al usuario la acción a seguir.
- La información debe ser clara y evitar ambigüedades.
- Se Debe utilizar un lenguaje acorde al usuario.

Reglas para la columna: TIPO_PROCESO

Esta columna debe contener el nombre de la funcionalidad a la cuál pertenece el mensaje, Ejemplos:

- Disponibilidad
- Compromisos
- Pago Automático
- Centros de Costos
- Cuentas por Cobrar

Reglas para la columna: CODIGO

Esta columna representa un código en texto del mensaje. Tener presente las siguientes consideraciones para la generación de los código externos de los mensajes:

Identificador

Debe empezar con el identificador "SICOF" en mayúsculas seguido por el número de la aplicación de la tabla SICOF.MAE_APLICACIONES columna CODIGO¹⁾, no debe tener espacios ni separadores al comienzo e intermedio del texto y debe finalizar con el separador _ Ejemplos:

Módulo	Identificador
FUNCIONALIDADES TRANSVERSALES	SICOF000_
SISTEMA DE CONTROL PRESUPUESTAL	SICOF001_
SISTEMA DE CONTROL DE TESORERIA	SICOF002_
SISTEMA DE CONTROL DE CONTABLE	SICOF003_
SISTEMA DE COMPRAS	SICOF004_
SISTEMA DE NOMINA	SICOF005_
SISTEMA DE TALENTO HUMANO	SICOF010_

Regla Nombre

El nombre del código del error debe ser un resumen del mensaje que se adiciona al identificador, este (si aplica) solo puede llevar el separador _ y se deben evitar los caracteres especiales y tildes. Tener

en cuenta las siguientes consideraciones:

- Nombre [nemotécnico](#)²⁾ del error³⁾.
- Resumen del error⁴⁾.
- Palabras clave del error⁵⁾.

Ejemplo

Para el mensaje “No existe el tercero con el nit.” se crearía el CODIGO de la siguiente forma:

- Identificador: **SICOF000_** Ya que es una funcionalidad transversal.
- Nombre: **TERCERO_NOT_FOUND**⁶⁾

Mensaje: No existe el tercero con el nit. Código de Error: SICOF000_TERCERO_NOT_FOUND

Ejemplo Script de Inserción

```
--
=====
==
-- Fecha: 10:03 a. m. lunes, 30 de agosto de 2021 - carlos.torres@ada.co
-- Crear Mensaje de Error para validar existencia del tercero.
--
=====
==
INSERT INTO PRESUP01.MENSAJES_SISTEMA
(CODIGO_MENSAJE, MENSAJE, INFORMATION, STOP, EXCLAMATION,
QUESTION, NONE, OK, CANCEL, YES,
NO, RETRY, ABORT, IGNORE, DEFAULTBUTTON,
CODIGO_MEMPRESA, FECHA_REGISTRO, TIPO_PROCESO, CODIGO,
TITULO)
VALUES
((SELECT NVL(MAX(CODIGO_MENSAJE), 0) + 1 FROM
PRESUP01.MENSAJES_SISTEMA),
'No existe información del tercero con el Nit #1', 'N', 'S', 'N',
'N', 'N', 'S', 'N', 'N',
'N', 'N', 'N', 'N', 1,
'999999999', SYSDATE, 'Tercero', 'SICOF000_TERCERO_NOT_FOUND',
'Validación del Tercero');
COMMIT;
```

Modo de uso: Powerbuilder - Documentación

Para visualizar la documentación debe descargar el siguiente repositorio [Documentación](#), abrir la pagina Index.html en su navegador web la cual es similar a la siguiente imagen:



En ella encontrará la documentación de las librerías que hacen parte del framework **Objetos SICOF** el cuál se irá actualizando frecuentemente a medida que se documenten las clases.

La Librería que contiene la funcionalidad del API de mensajes es la librería **sf00util.pbl**

Los Objetos relacionados en el API son:

- **n_cst_app**: Clase contenedora del API de mensajes
- **n_cst_msg**: Clase utilizada para la gestión de los mensajes que se visualizan en las aplicaciones.

Ejemplos de Uso

Para facilitar la implementación y uso del API de gestión de mensajes se crea un objeto interno privado en la clase global **guo_app** el cual puede ser accedido por el método **of_msg()** que devuelve la instancia del objeto. Sin embargo para implementaciones específicas se puede optar por crear y administrar la clase de mensajes **n_cst_msg** según considere el desarrollador.

A continuación se listan ejemplos de uso el cuál presenta las forma de utilizar el API, para más información debe consultar la documentación en el repositorio.

```
/*Ejemplos de uso utilizando la instancia genérica de la clase guo_app*/
guo_app.of_msg( ).of_msg_advertencia("Esto es una Advertencia.")
guo_app.of_msg( ).of_msg_informacion("Esto es una Información.")
guo_app.of_msg( ).of_msg_error("Esto es un Error.")

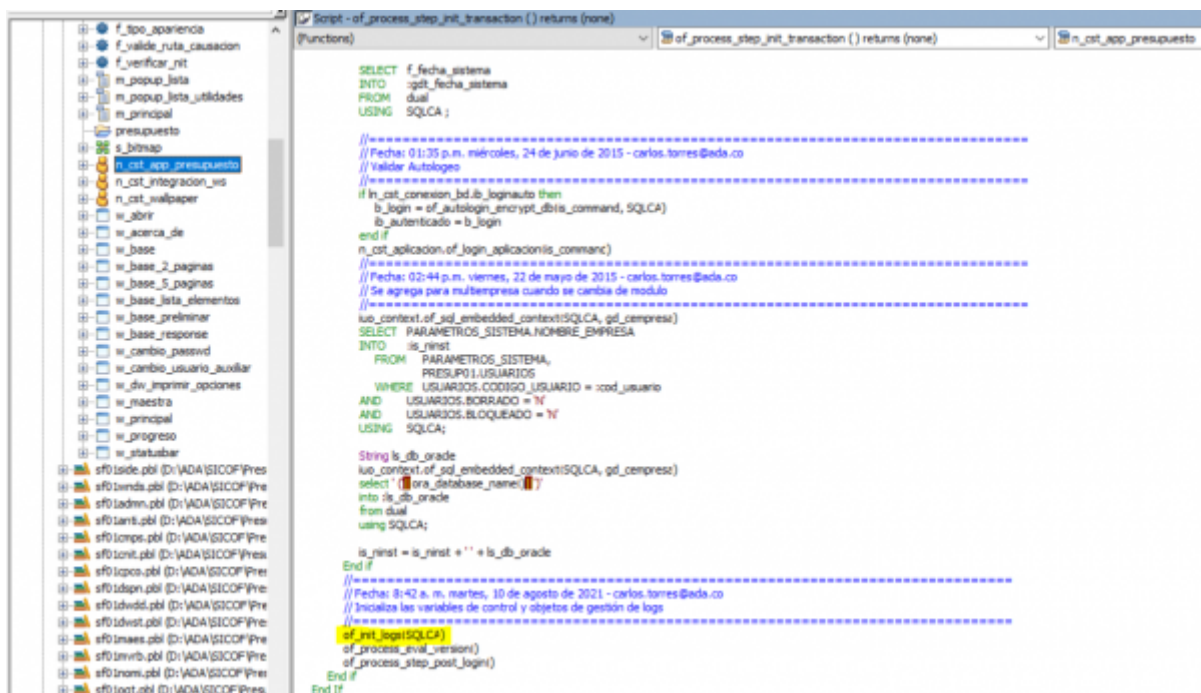
/*Ejemplo de uso utilizando el API de mensajes*/
guo_app.of_msg( ).of_mensajes_sistema(10)
guo_app.of_msg( ).of_mensajes_sistema(10, "Error al insert el encabezado")
guo_app.of_msg( ).of_mensajes_sistema('SICOF000_TERCERO_NOT_FOUND')

/*Ejemplo de uso definiendo la clase de mensaje.*/
n_cst_msg luomsg
luomsg = Create n_cst_msg
```

```
luo_msg.of_msg_advertencia("Esto es una Advertencia.")
luo_msg.of_msg_informacion("Esto es una Información.")
luo_msg.of_msg_error("Esto es un Error.")
luo_msg.of_mensajes_sistema(10)
luo_msg.of_mensajes_sistema(10, "Error al insert el encabezado")
luo_msg.of_mensajes_sistema('SICOF000_TERCERO_NOT_FOUND')
destroy luo_msg
```

Consideraciones

- El API puede ser activada o desactivada por medio de la constante: **API_MESSAGE** (Solo en aplicaciones SICOF ERP (Appeon/Powerbuilder)) siempre y cuando se utilice la implementación de la clase **guo_app**.
- El desarrollador es el encargado de gestionar la transacción que realiza las consultas e inicialización del objeto.
- Se unifica el desarrollo para que la función global **f_mensajes_sistema** tome los nuevos cambios sin afectar su definición.
- Cada módulo (Contabilidad, Presupuesto, Tesorería, Compras, Talento y Nómina) debe implementar el método de inicialización **guo_app.of_init_logs(SQLCA)** en el método **of_process_step_init_transaction** de la clase **guo_app** especializada por cada módulo. A continuación se muestra una imagen de referencia de la implementación del módulo de presupuesto. Utilice esta guía para implementaciones en otros módulos teniendo presente que la clase **n_cst_app** se especializa con el nombre de la aplicación que la contiene. Ejemplo: en presupuesto la clase especializada es **n_cst_app_presupuesto**, por lo general la clase esta en la libreria principal que contiene el objeto **Application**.



Modo de uso: Java

Para las aplicaciones desarrolladas en las tecnologías (Web):

- Java
- .Net
- PHP

el log de sesión será implementado por medio de un [Servicio Web](#) el cual deberá considerar las reglas de [Columna: WS](#)

[←Volver atras](#)

1)

Si el código es menor a 3 dígitos se deben rellenar con ceros a la izquierda. Ejemplo 1 = 001

2)

Un código mnemotécnico (o código nemotécnico) es un sistema sencillo utilizado para recordar una secuencia de datos, nombres, números, y en general para recordar listas de items que no pueden recordarse fácilmente.

3) , 4) , 5)

Puede estar en inglés

6)

no existe

From:

<http://wiki.adacsc.co/> - **Wiki**

Permanent link:

<http://wiki.adacsc.co/doku.php?id=ada:howto:sicoferp:factory:logmodels:message>

Last update: **2021/09/20 12:47**

