

## Guía de Correo.dll

### **Rutinas de TCI para la creación y recepción de mensajes**

A continuación se explica brevemente, las rutinas que se encuentran en el DLL, Correo.dll.

#### **Rutina para la creación de mensajes makeMSG2() :**

Esta rutina se encarga de crear los mensajes que se van a enviar vía el SAT, estos mensajes se generan con la extensión \*.MS2. Retorna mayor igual que **1**, si se creo el mensaje correctamente y **0**, si no se pudo crear.

#### **Ejemplo:**

```
Dim lngResultado as long  
lngResultado= makeMSG2 (strMailbox, strRutaArchivo & "\" & strArchivoZIPadjunto,  
UCase(strCasillaOrigen), UCase(strCasillaDestino), strClave, 1, 2)
```

Donde:

lngResultado devuelve el nombre del archivo MS2 generado.

```
strMailbox = Ruta Mailbox (c:\tci\mailbox)  
strRutaArchivo =Ruta archivo zip  
strArchivoZIPadjunto =NombrearchivoZIP  
strCasillaOrigen =NOMBRE DE CASILLA SAT DE LA AGENCIA DE ADUANA  
strCasillaDestino = SUNADTELMAR01 o SUNADTELAER01, etc. (casilla de la aduana de destino)  
strClave= #@SUNAD*&%
```

#### **Rutina para la recepción de mensajes recibMSG2() :**

Esta rutina se encarga de obtener los archivos adjuntos de los mensajes que le llegan vía el SAT, estos mensajes se reciben con la extensión \*.IN2. Retorna la cantidad de mensajes procesados.

#### **Ejemplo:**

```
Dim lngResultado as long  
lngResultado = recibMSG2(strMailbox & "\", strRutaAdjuntos & "\", 0, strClave)
```

Donde:

lngResultado devuelve la cantidad de mensajes desempaquetados.

```
strMailbox = Ruta carpeta mailbox (c:\tci\mailbox)  
strRutaAdjuntos = Ruta donde desempaquetara los adjuntos.  
strClave = #@SUNAD*&%
```

## Guía de Correo.dll

Código para declarar las funciones de la librería:

`Declaración de funciones para la administración de mensajes

### **Genera Mensaje**

Declare Function makeMSGA2 Lib "Correo.dll" (ByVal c\_pathMBOX As String, ByVal c\_pathFILE As String, ByVal c\_boxSOURCE As String, ByVal c\_boxDEST As String, ByVal c\_KEY As String, ByVal l\_overWRITR As Integer, ByVal l\_FLAGS As Integer) As long

### **Recibe Mensaje**

Declare Function recibMSG2 Lib "Correo.dll" (ByVal c\_pathMBOX As String, ByVal c\_pathREC As String, ByVal n\_CRITER As Integer, ByVal c\_KEY As String) As Integer

### **Ejemplo de invocación de las rutinas desde Visual Basic para generación de mensajes:**

En esta sección veremos lo que necesita para utilizar las rutinas dentro de Visual Basic.

1. Copiar el archivo Correo.Dll, en el directorio \WINDOWS\SYSTEM32\.
2. Crear un nuevo proyecto, dentro del cual se creará un *archivo de procedimientos* (*Module.Bas*).

Declarar dentro del *archivo de procedimientos* (*Module.Bas*) en la sección *General*, las rutina de la siguiente manera:

Declare Function makeMSGA2 Lib "Correo.dll" (ByVal c\_pathMBOX As String, ByVal c\_pathFILE As String, ByVal c\_boxSOURCE As String, ByVal c\_boxDEST As String, ByVal c\_KEY As String, ByVal l\_overWRITR As Integer, ByVal l\_FLAGS As Integer) As long

### **Ejemplo de invocación de las rutinas desde Visual Basic.NET para generación de mensajes:**

En esta sección veremos lo que necesita para utilizar las rutinas dentro de Visual Basic.

1. Copiar el archivo Correo.Dll, en el directorio \WINDOWS\SYSTEM32\.
3. Crear un nuevo proyecto, dentro del cual se creará un *archivo de procedimientos* (*Module.vb*).

Declarar dentro del *archivo de procedimientos* (*Module.vb*) en la sección *General*, las rutina de la siguiente manera:

## Guía de Correo.dll

Module CorreoDLL

```
Declare Function makeMSGa2 Lib "Correo.dll" (ByVal c_pathMBOX As String,  
ByVal c_pathFILE As String, ByVal c_boxSOURCE As String, ByVal c_boxDEST As  
String, ByVal c_KEY As String, ByVal l_overWRITR As Integer, ByVal l_FLAGS As  
Integer) As Long
```

End Module