



Av. San Juan de la Cruz, 31, 2ª, 6ºE
Telf.º : 968- 351364
Fax: 968- 345468
30011 – MURCIA

DOCUMENTO TÉCNICO INFORMATIVO

Componente: **ENVIOSMSAPPL.DLL**
Versión: **2.0**

miércoles, 16 de julio de 2008
FORO digital S.L.

Tel : 902 930 226
Mail: comercial@alertasonline.com
www.alertasonline.com



Av. San Juan de la Cruz, 31, 2ª, 6ºE
Telf.º : 968- 351364
Fax: 968- 345468
30011 – MURCIA

HISTORIAL DE VERSIONES

Fecha	Versión	Comentarios	Autor
15-03-2007	1.0	Documento Inicial	Juan Antonio
15-06-2007	1.1	Revisiones menores	Juan Antonio
15-09-2007	1.2	Revisiones menores	Juan Antonio
15-12-2007	1.3	Revisiones menores	Juan Antonio
15-03-2008	1.4	Revisiones menores	Juan Antonio
16-06-2008	1.5	Cambio de aspecto, integración con Delphi.	Juan Antonio
24-06-2008	2.0	Incluimos método SaldoCliente	Juan Antonio

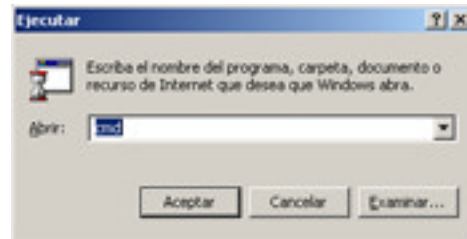
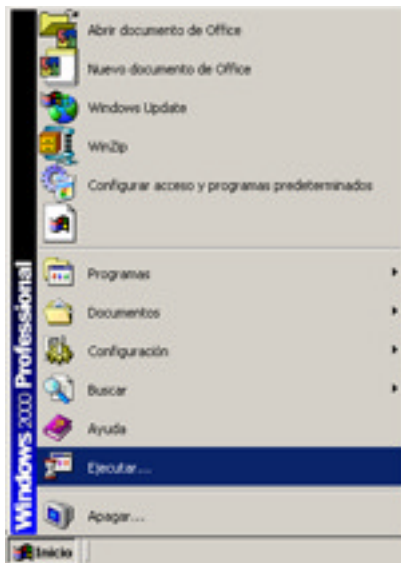
COMPONENTE ENVIOSMSAPPL.DLL

Antes de comenzar

Instalamos el paquete de Microsoft, que contiene: MSXML 4.0 Service Pack 2 (Servicios principales de Microsoft XML)
Archivo: msxmlspa-1.exe

¿Cómo registrar el componente?

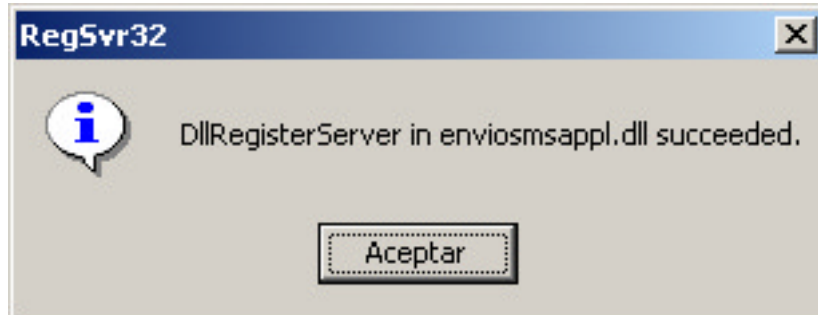
1. Copiamos el componente en una carpeta (con una localización viable y fácil de recordar).
Ejemplo:
C:\smsenvioappl
2. Inicio → Ejecutar... escribimos el comando “cmd”



3. Aparece la ventana de consola de Windows.
Nos posicionamos en la carpeta que hemos creado en el punto 1. Escribimos el siguiente comando para el registro del componente.

```
c:\smsenvioappl> regsvr32 enviosmsappl.dll.
```

Este comando nos devuelve el siguiente mensaje en pantalla:



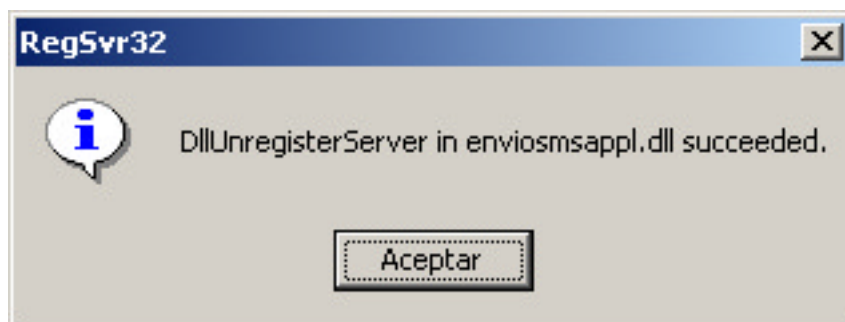
El componente queda registrado en el sistema

4. Si necesitamos quitar el registro del componente, seguimos los pasos 2 y 3. Ejecutamos el mismo comando añadiendo /u al final del mismo.

Ejemplo:

```
c:\smsenvioappl>regsvr32 enviosmsappl.dll /u
```

Este comando nos devuelve el siguiente mensaje en pantalla:



El componente deja de estar registrado en el sistema.

¿ Cómo actualizar el componente ?

Para actualizar el componente siga los siguientes pasos:

- 1.- Desregistrar el componente anterior mediante el siguiente comando:

```
c:\smsenvioappl>regsvr32 enviosmsappl.dll /u
```

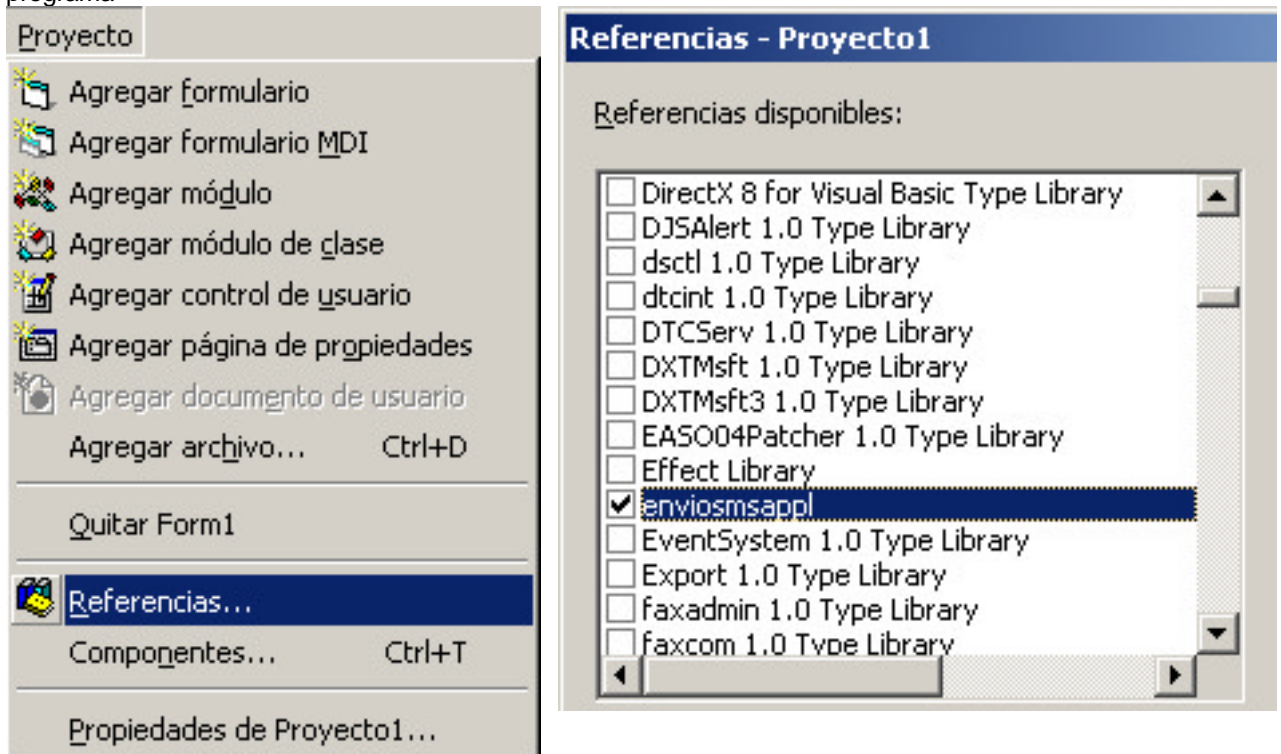
- 2.- Copiar el componente nuevo sobrescribiendo el anterior mediante el explorador de Windows o como le sea más cómodo.

- 3.- Registrar el componente nuevo

```
c:\smsenvioappl>regsvr32 enviosmsappl.dll
```

¿Cómo hacer referencia al objeto desde Visual Basic?

1. Abrimos Microsoft Visual Basic.
2. Hacemos clic en Proyectos → Referencias.
3. Buscamos el componente (enviosmsappl) dentro del listado de referencias que nos ofrece el programa



Con estos pasos, el programa ya puede utilizar el componente para el envío de información

¿Cómo hacer referencia al objeto desde Delphi?

Primero tenemos que importar la librería. Esto se hace desde el menú PROJECT -> IMPORT TYPE LIBRARY.

Una vez importada la librería, podremos hacer referencia a ella desde el código.
Ejemplo:



Av. San Juan de la Cruz, 31, 2ª, 6ºE
Telf.º : 968- 351364
Fax: 968- 345468
30011 – MURCIA

```
Procedure enviar_sms;  
var  
  mensaje : TSmsEnvio;  
  empresa,usuario,pass,texto,telefono : OleVariant;  
  Resultado : Integer;  
begin  
  Empresa := 'EmpresaPruebas';  
  usuario := 'UsuarioPruebas';  
  pass := 'PasswordPruebas';  
  texto := 'Prueba de mensaje';  
  telefono := '609xxxxxx';  
  mensaje := Tsmsenvio.Create(nil);  
  try  
    Resultado := mensaje.Enviosms(empresa,usuario,pass,texto,telefono);  
  finally  
    mensaje.free;  
  end  
end  
end
```

¿Cómo funciona?

DLL generada en Visual Basic, para el envío de SMS y demás operaciones mediante la plataforma SOAP de FORODIGITAL, S.L.

Métodos

- EnvioSMS (Envío de mensajes)
- Tracking (Recepción de estado del mensaje)
- SaldoCliente (Obtención de los créditos disponibles)

EnvioSMS

```
Respuesta = Objeto.EnvioSMS ("Empresa", "Usuario", "Pass", "Mensaje",  
"Numeros")
```

Empresa: Nombre de empresa (facilitado por Foro Digital).

Usuario: Usuario (facilitado por Foro Digital).

Pass: Password (facilitado por Foro Digital).

Mensaje: Mensaje a enviar

Numeros: Colección de números de terminales a los que se desea enviar el mensaje, separados por la coma “,”.

Ejemplo:

```
Res = Objeto.EnvioSMS("Empresa","Usuario","Pass","Esto es un mensaje de  
bienvenida","669XXXXXX,626XXXXXX")
```



Av. San Juan de la Cruz, 31, 2ª, 6ºE
Telf.º : 968- 351364
Fax: 968- 345468
30011 – MURCIA

En caso de ocurrir un error en la información facilitada al componente, éste devuelve un código numérico negativo:

Código	Mensaje de Error
-1000	Falta dato Empresa
-1001	Falta dato Usuario
-1002	Falta dato Password
-1003	Falta dato Mensaje
-1004	Mensaje incorrecto (contiene caracteres no válidos)
-1005	Falta dato Numeros
-1006	Error en número de teléfono
-3000	Error en la conexión.
-50000	Falta de crédito

Si la información es correcta, debe pasar por el filtro de nuestro SOAP, el resultado se devuelve en formato XML:

En caso de Error:

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <EnviarSmsResponse xmlns="http://forodigital.es/GestionSms/">
      <EnviarSmsResult>-3000</EnviarSmsResult>
    </EnviarSmsResponse>
  </soap:Body>
</soap:Envelope>
```

En caso de envío correcto:

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <EnviarSmsResponse xmlns="http://forodigital.es/GestionSms/">
      <EnviarSmsResult>551</EnviarSmsResult>
    </EnviarSmsResponse>
  </soap:Body>
</soap:Envelope>
```

El código obtenido es el id de mensaje que utilizaremos en la función Tracking para obtener el resultado de nuestro envío.

Tracking



Av. San Juan de la Cruz, 31, 2ª, 6ºE
Telf.º : 968- 351364
Fax: 968- 345468
30011 – MURCIA

```
Respuesta = Objeto.Tracking ("Empresa", "Usuario", "Pass", "Codigo")
```

Empresa: Nombre de empresa
Usuario: Usuario
Pass: Password
Codigo: Id de mensaje obtenido del envío.

Ejemplo:

```
Res = Objeto.Tracking("Empresa","Usuario","Pass","551")
```

En caso de ocurrir un error en la información facilitada al componente, devuelve un código numérico negativo:

Código	Mensaje
-1000	Falta dato Empresa
-1001	Falta dato Usuario
-1002	Falta dato Password
-1009	Falta Id de mensaje
-3000	Error en la conexión.

Si la información es correcta, debe pasar por el filtro de nuestro SOAP, el resultado nos es devuelto en formato XML:

En caso de Error:

```
<?xml version="1.0" encoding="utf-8"?>  
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"  
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xmlns:xsd="http://www.w3.org/2001/XMLSchema">  
  <soap:Body>  
    <EstadoSmsResponse xmlns="http://forodigital.es/GestionSms/" />  
  </soap:Body>  
</soap:Envelope>
```

Si el resultado es vacío se ha producido un error

En caso de envío correcto:

```
<?xml version="1.0" encoding="utf-8"?>  
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"  
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xmlns:xsd="http://www.w3.org/2001/XMLSchema">  
  <soap:Body>  
    <EstadoSmsResponse xmlns="http://forodigital.es/GestionSms/">  
      <EstadoSmsResult>  
        <ArrayOfString>  
          <string>669XXXXXX</string>  
        </ArrayOfString>  
      </EstadoSmsResult>  
    </EstadoSmsResponse>  
  </soap:Body>  
</soap:Envelope>
```




Av. San Juan de la Cruz, 31, 2ª, 6ºE
 Telf.º : 968- 351364
 Fax: 968- 345468
 30011 – MURCIA

```

    <string>Codigo</string>
  </ArrayOfString>
  <ArrayOfString>
    <string>626XXXXXX</string>
    <string>Codigo</string>
  </ArrayOfString>
</EstadoSmsResult>
</EstadoSmsResponse>
</soap:Body>
</soap:Envelope>

```

Los códigos del tracking se describen en esta tabla:

Código	Significado
0	Entregado al usuario.
10	Teléfono apagado o fuera de cobertura..
20	Entregado a la operadora. Entrega al usuario no confirmada.
40	La operadora ha recibido el mensaje y lo ha rechazado.
50	Rechazado por la plataforma por falta de crédito.
60	Rechazado por la plataforma.
70	Pendiente de envío en la plataforma.

El mensaje ha sido enviado pero aun no se ha obtenido el tracking de cada una de los terminales a los que le fue enviado.

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <EstadoSmsResponse xmlns="http://forodigital.es/GestionSms/">
      <EstadoSmsResult>
        <ArrayOfString>
          <string>669XXXXXX</string>
          <string />
        </ArrayOfString>
        <ArrayOfString>
          <string>626XXXXXX</string>
          <string />
        </ArrayOfString>
      </EstadoSmsResult>
    </EstadoSmsResponse>
  </soap:Body>
</soap:Envelope>

```



Av. San Juan de la Cruz, 31, 2ª, 6ºE
Telf.º : 968- 351364
Fax: 968- 345468
30011 – MURCIA

SaldoCliente

```
Respuesta = Objeto.SaldoCliente("Empresa","Usuario","Password")
```

Empresa: Nombre de la Empresa (facilitado por Foro Digital)
Usuario: Nombre del usuario (facilitado por Foro Digital)
Password: Contraseña del usuario (facilitado por Foro Digital)

Si no se ha producido ningún error, el método devuelve el crédito disponible en formato XML. Un ejemplo de una respuesta correcta sería:

```
<?xml version="1.0" encoding="utf-8"?>  
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"  
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xmlns:xsd="http://www.w3.org/2001/XMLSchema">  
  <soap:Body>  
    <SaldoClienteResponse xmlns="http://forodigital.es/GestionSms/">  
      <SaldoClienteResult>520</SaldoClienteResult>  
    </SaldoClienteResponse>  
  </soap:Body>  
</soap:Envelope>
```

En el caso de producirse un error, el método devuelve el código de error en formato XML. Un ejemplo de una respuesta con error sería:

```
<?xml version="1.0" encoding="utf-8"?>  
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"  
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xmlns:xsd="http://www.w3.org/2001/XMLSchema">  
  <soap:Body>  
    <SaldoClienteResponse xmlns="http://forodigital.es/GestionSms/">  
      <SaldoClienteResult>-1100</SaldoClienteResult>  
    </SaldoClienteResponse>  
  </soap:Body>  
</soap:Envelope>
```

Los códigos de error se describen en esta tabla:

Código	Mensaje
-1000	Falta dato Empresa.
-1001	Falta dato Usuario.
-1002	Falta dato Password.
-1003	Error de autenticación.
-1004	Error obteniendo el saldo.
-1100	Facturación mensual.
-3000	Error en la conexión.