

CONTPAQi® SDK

Asignar la dirección al cliente

En esta nota técnica encontrarás el procedimiento para generar recibo electrónico de pagos por medio de las funciones del SDK.

CONTPAQi®
Software empresarial fácil y completo

Aviso de derechos del propietario

Este Manual es una Obra Literaria protegida en favor de Computación en Acción, S.A. de C.V.; Copyright © 2004 Derechos Reservados © 2004 Computación en Acción, S.A. de C.V., Pablo Villaseñor No. 435, Col. Ladrón de Guevara, Guadalajara, Jalisco, México. C.P. 44600. Los Derechos de este Manual se encuentran reconocidos por la Ley Federal del Derecho de Autor. Se prohíbe su producción, reproducción, publicación, edición o fijación material en copias o ejemplares, por cualquier medio, importación, almacenamiento, transporte, distribución, comercialización, venta o arrendamiento, así como su comunicación y transmisión pública por cualquier medio, su divulgación en cualquier modalidad, su traducción, adaptación, paráfrasis, arreglos, transformaciones u otras similares, sin previa autorización por escrito de su titular. La violación de esta prohibición constituyen un delito y una infracción administrativa que están sancionados conforme a los artículos 424 fracción III, 424 bis fracción I y 424 ter, del Código Penal Federal; así como los artículos 229 fracciones VII y XVI y 231 fracciones I, III, IV y X, de la Ley Federal del Derecho de Autor y demás normas aplicables vigentes.

Las marcas COMPUTACIÓN EN ACCIÓN ®, EN ACCIÓN ®, PAQ ® y sus respectivos diseños; la marca y nombre comercial COMPAC ® y su diseño; las marcas ES TIEMPO DE PODER ®, LA CONEXIÓN DE TU NEGOCIO ®, TU NEGOCIO SIEMPRE EN MARCHA ®, SOÑAR. PODER. CRECER. ®; los avisos comerciales "Bien Pensado" ®, "Respuesta Oportuna" ®, y "La Forma más Amigable de Controlar tu Negocio" ®; así como la Imagen del Foquito ®© y del Diseño de la Portada ®©, son signos distintivos registrados y protegidos propiedad de Computación en Acción, S.A. de C.V.

AdminPAQ ®©, MegaPAQ ®©, Exión ®©, ContPAQ ®©, CheqPAQ ®©, NomiPAQ ®©, WinPAQ ®©, Solución Contable PAQ ®© y Ventpaq ®, también son marcas registradas y protegidas propiedad de Computación en Acción, S.A. de C.V., la que ostenta de igual forma los derechos patrimoniales de autor; con excepción del programa de cómputo que ostenta la marca VentPAQ, cuyos derechos patrimoniales pertenecen a Pacific Soft, S.A. de C.V.

Microsoft ®, MS-D.O.S. ®©, WINDOWS ®© y Excel ®©, son marcas y en su caso productos de Microsoft Corporation.

Cualquier otra marca que se mencione dentro de este manual que pertenezca a terceras partes tiene solamente propósitos informativos y no constituye aprobación y/o recomendación. Computación en Acción, no se responsabiliza de la ejecución o uso de estos productos.

Contenido

CONTPAQi ® SDK	1
SDK CONTPAQi ® Comercial	6
Estructuras	6
Documentos	6
RegLlaveDoc	7
Funciones	7
fAltaDocumentoCargoAbono	7
fEditarDocumento	7
fSetDatoDocumento.....	8
fGuardaDocumento.....	8
fSaldarDocumento	8
fEmitirDocumento.....	8
Ejemplo.....	9
SDK CONTPAQi ® Factura electrónica y AdminPAQ	12
Estructuras	12
Documentos	12
RegLlaveDoc	13
Funciones	13
fAltaDocumentoCargoAbono	13
fEditarDocumento	13
fSetDatoDocumento.....	14
fGuardaDocumento.....	14
fAfectaDocto_Param	14
fSaldarDocumento	14
fInicializaLicenseInfo	15
fEmitirDocumento.....	15
Ejemplo.....	16
Timbrado de XML	18
fInicializaLicenseInfo	18

fTimbraComplementoPagoXML.....	18
Ejemplo.....	18
Cancelar pagos	19
fInicializaLicenseInfo	19
fCancelaComplementoPagoUUID	19
Ejemplo.....	19
Vínculos de apoyo	20

CONTPAQ *i*[®]

Comercial

SDK CONTPAQi® Comercial

Estructuras

Documentos

```
[StructLayout(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack = 4)]  
public struct tDocumento  
{  
  
    public Double aFolio;  
    public int aNumMoneda;  
    public Double aTipoCambio;  
    public Double aImporte;  
    public Double aDescuentoDoc1;  
    public Double aDescuentoDoc2;  
    public int aSistemaOrigen;  
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = constantes.kLongCodigo)]  
    public String aCodConcepto;  
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = constantes.kLongSerie)]  
    public String aSerie;  
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = constantes.kLongFecha)]  
    public String aFecha;  
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = constantes.kLongCodigo)]  
    public String aCodigoCteProv;  
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = constantes.kLongCodigo)]  
    public String aCodigoAgente;  
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = constantes.kLongReferencia)]  
    public String aReferencia;  
    public double aAfecta;  
    public double aGasto1;  
    public double aGasto2;  
    public double aGasto3;  
}
```

RegLlaveDoc

```
[StructLayout(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack = 4)]  
public struct RegLlaveDoc  
{  
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = constantes.kLongCodigo)]  
    public String aCodConcepto;  
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = constantes.kLongSerie)]  
    public String aSerie;  
    public double folio;  
}
```

Nota: Esta estructura debe ser instanciada dos veces, ya que esta estructura será utilizada para escribir los valores del documento a saldar y del documento de pago generado.

Funciones

fAltaDocumentoCargoAbono

```
[DllImport("MGWServicios.dll")]  
public static extern int fCuentaBancariaEmpresaDoctos(string aCuentaBancaria);
```

Esta función nos ayuda a asignar la cuenta bancaria de la empresa, y debe ser utilizada antes de generar el documento.

fAltaDocumentoCargoAbono

```
[DllImport("MGWServicios.DLL")]  
public static extern Int32 fAltaDocumentoCargoAbono(ref tDocumento atDocumento);
```

Esta función generará el encabezado del documento en base a los datos proporcionados en la estructura.

fEditarDocumento

```
[DllImport("MGWServicios.DLL")]  
public static extern Int32 fEditarDocumento();
```

Esta función nos ayudara a poner en modo edición el registro actual en la tabla MGW10008 (Documentos).

fSetDatoDocumento

```
[DllImport("MGWServicios.DLL")]  
public static extern Int32 fSetDatoDocumento(string aCampo, string aValor);
```

Esta función nos ayudará a ingresar información en algún campo indicado por el parámetro aCampo, esta información se verá reflejado en la tabla MGW10008 (Documentos).

fGuardaDocumento

```
[DllImport("MGWServicios.DLL")]  
public static extern Int32 fGuardaDocumento();
```

Esta función guarda los cambios realizados por la función fSetDatoDocumento

fSaldarDocumento

```
[DllImport("MGWServicios.DLL")]  
public static extern Int32 fSaldarDocumento(ref RegLlaveDoc astDocAPagar,  
    ref RegLlaveDoc astDocPago, double importe, int moneda, string aFecha);
```

Esta función nos ayudará a saldar o abonar el importe indicado al documento deseado, esta información se pasará por medio de los parámetros de la función.

fEmitirDocumento

```
[DllImport("MGWServicios.DLL")]  
public static extern Int32 fEmitirDocumento([MarshalAs(UnmanagedType.LPStr)] string aCodConcepto,  
    [MarshalAs(UnmanagedType.LPStr)] string aSerie, double aFolio,  
    [MarshalAs(UnmanagedType.LPStr)] string aPassword,  
    [MarshalAs(UnmanagedType.LPStr)] string aArchivoAdicional);
```

Esta función nos ayudara a realizar el timbrado del documento de pago.

Ejemplo

```

SDK.lError = SDK.fCuentaBancariaEmpresaDoctos("1234567890123456");
double folio = 0;
SDK.tDocumento DoctoCargoAbono = new SDK.tDocumento();
SDK.RegLlaveDoc saldar = new SDK.RegLlaveDoc();
SDK.RegLlaveDoc pago = new SDK.RegLlaveDoc();

var concepto = seleccionconcepto(cmbConceptos.SelectedItem.ToString());
SDK.lError = SDK.fSiguienteFolio(concepto, serie, ref folio);

DoctoCargoAbono.aCodConcepto = concepto;
DoctoCargoAbono.aFolio = folio;
DoctoCargoAbono.aSerie = "";

DoctoCargoAbono.aFecha = DateTime.Today.ToString("MM/dd/yyyy");
DoctoCargoAbono.aCodigoCteProv = "CL001";
DoctoCargoAbono.aTipoCambio = 1;
DoctoCargoAbono.aNumMoneda = 1;
DoctoCargoAbono.aImporte = 116; //Importe total del documento de pago
DoctoCargoAbono.aReferencia = "DOCTO SDK";

SDK.lError = SDK.fAltaDocumentoCargoAbono(ref DoctoCargoAbono);
if (SDK.lError != 0)
{
    SDK.rError(SDK.lError);
}
else
{
    MessageBox.Show("Documento creado");
    SDK.fEditarDocumento();
    SDK.lError = SDK.fSetDatoDocumento("CMETODOPAG", "01"); //Forma de pago
    SDK.lError = SDK.fSetDatoDocumento("CCANTPARCI", "1"); //Cantidad de parcialidades
    SDK.lError = SDK.fSetDatoDocumento("CNUMPARCIA", "1"); //Numero de parcialidad
    SDK.lError = SDK.fSetDatoDocumento("CUSUBAN03", "09:10:00"); //Se asigna la hora del pago
    //Identificador de la cuenta bancaria del cliente
    //SDK.lError = SDK.fSetDatoDocumento("CIDMONEDCA", "2");
    SDK.lError = SDK.fGuardaDocumento();

    saldar.aCodConcepto = "42017"; //codigo de concepto del documento a saldar
    saldar.aSerie = ""; //Serie del documento a saldar
    saldar.folio = 2134631; //Folio del documento a saldar

    pago.aCodConcepto = concepto; //Concepto del documento de pago
    pago.aSerie = ""; //Serie del documento de pago
    pago.folio = folio; //Folio del documento de pago
  }

```

```
SDK.lError = SDK.fSaldarDocumento(ref saldar, ref pago, 116, 1,  
    DateTime.Now.ToString("MM/dd/yyyy"));  
if (SDK.lError != 0)  
{  
    SDK.rError(SDK.lError);  
}  
else  
{  
    SDK.lError = SDK.fEmitirDocumento(concepto, "", folio, "12345678a", "");  
    if (SDK.lError != 0)  
    {  
        SDK.rError(SDK.lError);  
    }  
    else  
    {  
        MessageBox.Show("Documento timbrado");  
    }  
}  
}
```

AdminPAQ®

CONTPAQ *i*®

Factura electrónica

SDK CONTPAQi® Factura electrónica y AdminPAQ

Estructuras

Documentos

```
[StructLayout(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack = 4)]
public struct tDocumento
{
    public Double aFolio;
    public int aNumMoneda;
    public Double aTipoCambio;
    public Double aImporte;
    public Double aDescuentoDoc1;
    public Double aDescuentoDoc2;
    public int aSistemaOrigen;
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = constantes.kLongCodigo)]
    public String aCodConcepto;
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = constantes.kLongSerie)]
    public String aSerie;
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = constantes.kLongFecha)]
    public String aFecha;
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = constantes.kLongCodigo)]
    public String aCodigoCteProv;
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = constantes.kLongCodigo)]
    public String aCodigoAgente;
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = constantes.kLongReferencia)]
    public String aReferencia;
    public double aAfecta;
    public double aGasto1;
    public double aGasto2;
    public double aGasto3;
}
```

RegLlaveDoc

```
[StructLayout(LayoutKind.Sequential, CharSet = CharSet.Ansi, Pack = 4)]  
public struct RegLlaveDoc  
{  
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = constantes.kLongCodigo)]  
    public String aCodConcepto;  
    [MarshalAs(UnmanagedType.ByValTStr, SizeConst = constantes.kLongSerie)]  
    public String aSerie;  
    public double folio;  
}
```

Nota: Esta estructura debe ser instanciada dos veces, ya que esta estructura será utilizada para escribir los valores del documento a saldar y del documento de pago generado.

Funciones

fAltaDocumentoCargoAbono

```
[DllImport("MGW_SDK.DLL")]  
public static extern Int32 fAltaDocumentoCargoAbono(ref tDocumento atDocumento);
```

Esta función generará el encabezado del documento en base a los datos proporcionados en la estructura.

fEditarDocumento

```
[DllImport("MGW_SDK.DLL")]  
public static extern Int32 fEditarDocumento();
```

Esta función nos ayudara a poner en modo edición el registro actual en la tabla MGW10008 (Documentos).

fSetDatoDocumento

```
[DllImport("MGW_SDK.DLL")]  
public static extern Int32 fSetDatoDocumento(string aCampo, string aValor);
```

Esta función nos ayudará a ingresar información en algún campo indicado por el parámetro aCampo, esta información se verá reflejado en la tabla MGW10008 (Documentos).

fGuardaDocumento

```
[DllImport("MGW_SDK.DLL")]  
public static extern Int32 fGuardaDocumento();
```

Esta función guarda los cambios realizados por la función fSetDatoDocumento

fAgregaHoraREP

```
[DllImport("MGW_SDK.DLL")]  
public static extern Int32 fAgregaHoraREP(string aCodConcepto, string aSerie, string aFolio, string aHora);
```

Esta función nos ayuda a asignar la hora del REP

fAfectaDocto_Param

```
[DllImport("MGW_SDK.DLL")]  
public static extern Int32 fAfectaDocto_Param([MarshalAs(UnmanagedType.LPStr)] string aCodConcepto,  
[MarshalAs(UnmanagedType.LPStr)] string aSerie, double aFolio, bool aAfecta);
```

Esta función nos ayudará a realizar la afectación de los documentos, esta afectación se realizará para los saldos del cliente, existencias del producto, etc.

fSaldarDocumento

```
[DllImport("MGW_SDK.DLL")]  
public static extern Int32 fSaldarDocumento(ref RegLlaveDoc astDocAPagar,  
ref RegLlaveDoc astDocPago, double importe, int moneda, string aFecha);
```

Esta función nos ayudará a saldar o abonar el importe indicado al documento deseado, esta información se pasará por medio de los parámetros de la función.

fInicializaLicenseInfo

```
[DllImport("MGW_SDK.DLL")]  
public static extern int fInicializaLicenseInfo(byte aSistema);
```

Esta función nos ayudará a leer la licencia del sistema.

fEmitirDocumento

```
[DllImport("MGW_SDK.DLL")]  
public static extern Int32 fEmitirDocumento([MarshalAs(UnmanagedType.LPStr)] string aCodConcepto,  
[MarshalAs(UnmanagedType.LPStr)] string aSerie, double aFolio,  
[MarshalAs(UnmanagedType.LPStr)] string aPassword,  
[MarshalAs(UnmanagedType.LPStr)] string aArchivoAdicional);
```

Esta función nos ayudara a realizar el timbrado del documento de pago.

Ejemplo

```
double folio = 0;

SDK.tDocumento DoctoCargoAbono = new SDK.tDocumento();
SDK.RegLlaveDoc saldar = new SDK.RegLlaveDoc();
SDK.RegLlaveDoc pago = new SDK.RegLlaveDoc();

var concepto = selecconcepto(cmbConceptos.SelectedItem.ToString());
SDK.lError = SDK.fSiguienteFolio(concepto, serie, ref folio);

DoctoCargoAbono.aCodConcepto = concepto;
DoctoCargoAbono.aFolio = folio;
DoctoCargoAbono.aSerie = "";

DoctoCargoAbono.aFecha = DateTime.Today.ToString("MM/dd/yyyy");
DoctoCargoAbono.aCodigoCteProv = "CL001";
DoctoCargoAbono.aTipoCambio = 1;
DoctoCargoAbono.aNumMoneda = 1;
DoctoCargoAbono.aImporte = 100;
DoctoCargoAbono.aReferencia = "DOCTO SDK";

SDK.lError = SDK.fAltaDocumentoCargoAbono(ref DoctoCargoAbono);
if (SDK.lError != 0)
{
    SDK.rError(SDK.lError);
}
else
{
    MessageBox.Show("Documento creado");
    SDK.fEditarDocumento();
    //Se asigna el Id de la cuenta bancaria de la empresa
    SDK.lError = SDK.fSetDatoDocumento("CIDCUENTA", "3");
    SDK.lError = SDK.fSetDatoDocumento("CMETODOPAG", "03");//Forma de pago
    SDK.lError = SDK.fSetDatoDocumento("CCANTPARCI", "1");//Cantidad de parcialidades
    SDK.lError = SDK.fSetDatoDocumento("CNUMPARCIA", "1");//Numero de parcialidad
    //Identificador de la cuenta bancaria del cliente
    SDK.lError = SDK.fSetDatoDocumento("CIDMONEDCA", "6");
    SDK.lError = SDK.fGuardaDocumento();

    SDK.lError = SDK.fAgregarHoraREP(concepto, "", folio.ToString(), "09:19:00");

    saldar.aCodConcepto = "42017";//codigo de concepto del documento a saldar
    saldar.aSerie = "";//Serie del documento a saldar
    saldar.folio = 2134826;//Folio del documento a saldar

    pago.aCodConcepto = concepto;//Concepto del documento de pago
    pago.aSerie = "";//Serie del documento de pago
    pago.folio = folio;//Folio del documento de pago
```



```
//Se realiza la afectación del documento
SDK.lError= SDK.fAfectaDocto_Param(concepto, "", folio, true);

SDK.lError = SDK.fSaldarDocumento(ref saldar, ref pago, 116, 1,
    DateTime.Now.ToString("MM/dd/yyyy"));
if (SDK.lError != 0)
{
    SDK.rError(SDK.lError);
}
else
{
    //Se lee la licencia de AdminPAQ
    SDK.fInicializaLicenseInfo(0);

    SDK.lError = SDK.fEmitirDocumento(concepto, "", folio, "12345678a", "");
    if (SDK.lError != 0)
    {
        SDK.rError(SDK.lError);
    }
    else
    {
        MessageBox.Show("Documento timbrado");
    }
}
}
```

Nota:

Recordar que es necesario realizar la afectación del documento antes de saldarlo.

Timbrado de XML

fInicializaLicenseInfo

```
[DllImport("MGW_SDK.DLL")]  
public static extern int fInicializaLicenseInfo(byte aSistema);
```

Esta función nos ayudará a leer la licencia del sistema.

*Esta licencia debe ser de 5 usuarios

fTimbraComplementoPagoXML

```
[DllImport("MGW_SDK.dll")]  
public static extern int fTimbraComplementoPagoXML(string aRutaXML,  
    string aCodConcepto, StringBuilder aUUID,  
    string aRutaDDA, string aRutaResultado, string aPass,  
    string aRutaFormato);
```

Esta función nos ayudará a realizar el timbrado del XML previamente generado.

Ejemplo

```
var concepto = selecconcepto(cmbConceptos.SelectedItem.ToString());  
SDK.lError = SDK.fInicializaLicenseInfo(0);  
StringBuilder UUID = new StringBuilder(300);  
  
SDK.lError = SDK.fTimbraComplementoPagoXML(@"C:\Compacw\Empresas\Admin\XML_RecepPagos\PagoUSD.xml",  
    concepto, UUID, "",  
    @"C:\Compacw\Empresas\Admin\Timbrados",  
    "12345678a",  
    @"C:\Compacw\Empresas\Reportes\Facturacion\Plantilla_Factura_cfdi_1.htm");  
  
if (SDK.lError != 0)  
{  
    SDK.rError(SDK.lError);  
}  
else  
{  
    MessageBox.Show($"UUID: {UUID.ToString()}");  
}
```

Cancelar pagos

fInicializaLicenseInfo

```
[DllImport("MGW_SDK.DLL")]  
public static extern int fInicializaLicenseInfo(byte aSistema);
```

Esta función nos ayudará a leer la licencia del sistema.

*Esta licencia debe ser de 5 usuarios

fCancelaComplementoPagoUUID

```
[DllImport("MGW_SDK.DLL")]  
public static extern Int32 fCancelaComplementoPagoUUID(string aUUID,  
string aIdDConcepto, string aPass);
```

Esta función nos ayudará a realizar la cancelación de un REP.

Ejemplo

```
string uuid = "6CBE2232-0856-4DA0-B636-E00EB8C6DDCF";  
  
string concepto = "3006";  
string pass = "12345678a";  
  
int lError = 0;  
lError = SDK.fInicializaLicenseInfo(0);  
lError = SDK.fCancelaComplementoPagoUUID(uuid, concepto, pass);  
if (lError != 0)  
{  
    SDK.rError(lError);  
}  
else  
{  
    MessageBox.Show("Documento Cancelado");  
}
```

Vínculos de apoyo

Guía de llenado para el complemento recepción de pagos:

http://omawww.sat.gob.mx/informacion_fiscal/factura_electronica/Documents/Complementoscfdi/Guia_comple_pagos.pdf

Cambios fiscales en los sistemas CONTPAQi

<https://www.contpaqi.com/cambiosfiscales/>