

Proyecto: Integración de Timbrado usando PHP
Documento: Manual de implementación

1. TimbraCFDI

```
<?php
/* Ruta del servicio de integracion*/
$ws = "http://www.timbracfdipruebas.mx/ServicioIntegracionPruebas/Timbrado.asmx?wsdl";
$response = '';
/* Ruta del comprobante a timbrar*/
$rutaArchivo = 'C:\comprobanteSinTimbrar.xml';
/* El servicio recibe el comprobante (xml) codificado en Base64, el rfc del emisor deberá ser 'AAA010101AAA'
para efecto de pruebas*/
$base64Comprobante = file_get_contents($rutaArchivo);
$base64Comprobante = base64_encode($base64Comprobante);
try
{
    $params = array();
    /*Nombre del usuario integrador asignado, para efecto de pruebas utilizaremos 'mvpNUXmQfK8='*/
    $params['usuarioIntegrador'] = 'mvpNUXmQfK8=';
    /* Comprobante en base 64*/
    $params['xmlComprobanteBase64'] = $base64Comprobante;
    /*Id del comprobante, deberá ser un identificador único, para efecto del ejemplo se utilizará un numero
    aleatorio*/
    $params['idComprobante'] = rand(5, 999999);
    $client = new SoapClient($ws,$params);
    $response = $client->__soapCall('TimbraCFDI', array('parameters' => $params));
}
catch (SoapFault $fault)
{
    echo "SOAPFault: ".$fault->faultcode."-".$fault->faultstring."\n";
}
/*Obtenemos resultado del response*/
$tipoExcepcion = $response->TimbraCFDIResult->anyType[0];
$numeroExcepcion = $response->TimbraCFDIResult->anyType[1];
$descripcionResultado = $response->TimbraCFDIResult->anyType[2];
$xmlTimbrado = $response->TimbraCFDIResult->anyType[3];
$codigoQr = $response->TimbraCFDIResult->anyType[4];
$cadenaOriginal = $response->TimbraCFDIResult->anyType[5];

if($xmlTimbrado != '')
{
    /*El comprobante fue timbrado correctamente*/

    /*Guardamos comprobante timbrado*/
    file_put_contents('C:\comprobanteTimbrado.xml', $xmlTimbrado);

    /*Guardamos codigo qr*/
    file_put_contents('C:\codigoQr.jpg', $codigoQr);

    /*Guardamos cadena original del complemento de certificacion del SAT*/
    file_put_contents('C:\cadenaOriginal.txt', $cadenaOriginal);

    print_r("Timbrado exitoso");
}
else
{
    print_r($descripcionResultado);
}
?>
```

2. ObtieneCFDI

```
<?php
/* Ruta del servicio de integracion*/
$ws = "http://www.timbracfdipruebas.mx/ServicioIntegracionPruebas/Timbrado.asmx?wsdl";
$response = '';
/* El servicio para obtener un cfdi previamente timbrado recibe 3 parámetros*/

/*Usuario Integrador*/
$usuarioIntegrador = 'mvpNUXmQfK8=';

/*Rfc del Emisor que emitió el comprobante*/
$rfcEmisor = 'AAA010101AAA';

/*Folio fiscal(UUID) del comprobante a obtener, deberá ser uno válido de los que hayamos timbrado previamente en pruebas*/
$folioUUID = '2961B686-65A1-4A0D-8C76-76D95B6558DF';

try
{
    $params = array();
    /*Nombre del usuario integrador asignado, para efecto de pruebas utilizaremos 'mvpNUXmQfK8='*/
    $params['usuarioIntegrador'] = $usuarioIntegrador;
    /* Rfc emisor que emitió el comprobante*/
    $params['rfcEmisor'] = $rfcEmisor;
    /*Folio fiscal del comprobante a obtener*/
    $params['folioUUID'] = $folioUUID;

    $client = new SoapClient($ws,$params);
    $response = $client->__soapCall('ObtieneCFDI', array('parameters' => $params));
}
catch (SoapFault $fault)
{
    echo "SOAPFault: ".$fault->faultcode."-".$fault->faultstring."\n";
}

/*Obtenemos resultado del response*/
$tipoExcepcion = $response->ObtieneCFDIResult->anyType[0];
$numeroExcepcion = $response->ObtieneCFDIResult->anyType[1];
$descripcionResultado = $response->ObtieneCFDIResult->anyType[2];
$xmlTimbrado = $response->ObtieneCFDIResult->anyType[3];
$codigoQr = $response->ObtieneCFDIResult->anyType[4];
$cadenaOriginal = $response->ObtieneCFDIResult->anyType[5];

if($numeroExcepcion == "0")
{
    /*El comprobante fue consultado exitosamente*/

    /*Guardamos comprobante timbrado*/
    file_put_contents('C:\comprobanteConsultado.xml', $xmlTimbrado);

    print_r("El comprobante fue consultado correctamente");
}
else
{
    print_r($descripcionResultado);
}
?>
```

3. CancelaCFDI

```
<?php
/* Ruta del servicio de integracion*/
$ws = "http://www.timbracfdipruebas.mx/ServicioIntegracionPruebas/Timbrado.asmx?wsdl";
$response = '';
/* El servicio para cancelar un cfdi recibe 3 parámetros*/

/*Usuario Integrador*/
$usuarioIntegrador = 'mvpNUXmQfK8=';

/*Rfc del Emisor que emitió el comprobante*/
$rfcEmisor = 'AAA010101AAA';

/*Folio fiscal(UUID) del comprobante a cancelar, deberá ser uno válido de los que hayamos timbrado previamente
en pruebas*/
$folioUUID = '2961B686-65A1-4A0D-8C76-76D95B6558DF';

try
{
    $params = array();
    /*Nombre del usuario integrador asignado, para efecto de pruebas utilizaremos 'mvpNUXmQfK8='*/
    $params['usuarioIntegrador'] = $usuarioIntegrador;
    /* Rfc emisor que emitió el comprobante*/
    $params['rfcEmisor'] = $rfcEmisor;
    /*Folio fiscal del comprobante a cancelar*/
    $params['folioUUID'] = $folioUUID;

    $client = new SoapClient($ws,$params);
    $response = $client->__soapCall('CancelaCFDI', array('parameters' => $params));
}
catch (SoapFault $fault)
{
    echo "SOAPFault: ".$fault->faultcode."-".$fault->faultstring."\n";
}
/*Obtenemos resultado del response*/
$tipoExcepcion = $response->CancelaCFDIResult->anyType[0];
$numeroExcepcion = $response->CancelaCFDIResult->anyType[1];
$descripcionResultado = $response->CancelaCFDIResult->anyType[2];
$xmlTimbrado = $response->CancelaCFDIResult->anyType[3];
$codigoQr = $response->CancelaCFDIResult->anyType[4];
$cadenaOriginal = $response->CancelaCFDIResult->anyType[5];

if($numeroExcepcion == "0")
{
    /*El comprobante fue cancelado exitosamente*/

    print_r("El comprobante fue cancelado correctamente");
}
else
{
    print_r($descripcionResultado);
}
?>
```

4. RegistraEmisor

```
<?php
/* Ruta del servicio de integracion*/
$ws = "http://www.timbracfdipuebas.mx/ServicioIntegracionPruebas/Timbrado.asmx?wsdl";
$response = '';
/* El servicio para registrar un emisor recibe 5 parámetros*/

/*Usuario Integrador*/
$usuarioIntegrador = 'mvpNUXmQfK8=';

/*Rfc Emisor a registrar*/
$rfcEmisor = 'AAA010101AAA';

/*Archivo .cer base64*/
$rutaCer = 'C:\aaa010101aaa_csd_01.cer';
$base64Cer = file_get_contents($rutaCer);
$base64Cer = base64_encode($base64Cer);

/*Archivo .key base64*/
$rutaKey = 'C:\aaa010101aaa_csd_01.key';
$base64Key = file_get_contents($rutaKey);
$base64Key = base64_encode($base64Key);

/*Contraseña del sello*/
$contrasena = '12345678a';

try
{
    $params = array();
    /*Nombre del usuario integrador asignado, para efecto de pruebas utilizaremos 'mvpNUXmQfK8='*/
    $params['usuarioIntegrador'] = $usuarioIntegrador;
    /* Rfc emisor a registrar 64*/
    $params['rfcEmisor'] = $rfcEmisor;
    /*Archivo .cer en base 64, sello digital del emisor*/
    $params['base64Cer'] = $base64Cer;
    /*Archivo .key en base 64, sello digital del emisor*/
    $params['base64Key'] = $base64Key;
    /*Contraseña, sello digital del emisor*/
    $params['contrasena'] = $contrasena;
    $client = new SoapClient($ws,$params);
    $response = $client->__soapCall('RegistraEmisor', array('parameters' => $params));
}
catch (SoapFault $fault)
{
    echo "SOAPFault: ".$fault->faultcode."-".$fault->faultstring."\n";
}

/*Obtenemos resultado del response*/
$tipoExcepcion = $response->RegistraEmisorResult->anyType[0];
$numeroExcepcion = $response->RegistraEmisorResult->anyType[1];
$descripcionResultado = $response->RegistraEmisorResult->anyType[2];
$xmlTimbrado = $response->RegistraEmisorResult->anyType[3];
$codigoQr = $response->RegistraEmisorResult->anyType[4];
$cadenaOriginal = $response->RegistraEmisorResult->anyType[5];

if($numeroExcepcion == "0")
{
    /*El emisor fue registrado correctamente*/

    print_r("El emisor fue registrado correctamente");
}
else
{
    print_r($descripcionResultado);
}
?>
```