

SISTEMA DE EMISSÃO DE NOTAS FISCAIS ELETRÔNICAS DE SERVIÇOS

MANUAL BÁSICO DE INTEGRAÇÃO DO WEB SERVICE eNotaFiscal-WS

Versão 1.3
(PILOTO)

10/2019

Sumário

Apresentação.....	3
Lista de Mudanças.....	4
Processo de Credenciamento.....	5
Especificação.....	6
Operações do Web Service.....	7
CABEÇALHO DAS REQUISIÇÕES.....	7
GERAÇÃO DE NFS-E.....	7
CONSULTA DE NFS-E.....	8
RPS Eletrônico.....	9
Considerações Sobre o Contrato WSDL.....	11
ESPECIFICIDADES EM ALGUNS FORMATOS DE CAMPO.....	11
ESPECIFICIDADES EM ALGUNS CONTEÚDOS DE CAMPO.....	11
Lista de Erros e Alertas.....	12
LISTA DE ALERTAS DO MUNICÍPIO.....	12
LISTA DE ERROS DO MUNICÍPIO.....	12
Exemplos de códigos-fonte.....	13
FERRAMENTA SOAPUI.....	13
CONSUMO EM JAVA.....	15
CONSUMO EM PHP.....	18

Apresentação

Este manual destina-se ao esclarecimento de alguns pontos e à orientação básica quanto a integração de sistemas informatizados de terceiros ao módulo emissor de notas fiscais de serviço eletrônicas por meio do *web service* eNotaFiscal-WS.

Lista de Mudanças

Nesta seção demonstrar-se-á as mudanças aplicadas neste manual conforme a evolução das versões.

Versão 1.0 - PILOTO	* Primeira versão - não constam mudanças.
Versão 1.1 - PILOTO	* Inclusão das mensagens de erro personalizadas L109, L110, L111, L112 e L113; * Inclusão da mensagem de alerta personalizada AL003.
Versão 1.2 - PILOTO	* Inclusão da figura e complemento de orientações acerca do pedido de AIDF-e para RPS eletrônico.
Versão 1.3 - PILOTO	* Atualização do logotipo do Município no cabeçalho. * Inclusão de exemplos de códigos-fonte em JAVA e PHP.

Processo de Credenciamento

O credenciamento ao uso do sistema eNotaFiscal-WS está condicionado diretamente ao processo de credenciamento já deferido do sistema web eNotaFiscal e está disponível apenas para contribuintes pessoa jurídica.

Durante a fase denominada “Piloto” do *web service*, as empresas selecionadas e interessadas receberão visita do pessoal da Receita Municipal, momento no qual será formalizado o pedido de credenciamento. O contribuinte deverá informar um e-mail válido que servirá para receber as mensagens enviadas pelo sistema durante a fase de testes e, também, assinar o termo do serviço. Oportunamente, nesta fase, uma cópia deste manual será remetida por email.

Posteriormente, após o encerramento da fase Piloto, as novas solicitações deverão ser feitas via opção “*Configurações*” no menu do usuário do sistema web eNotaFiscal.

Após a realização do pedido, o contribuinte receberá, via e-mail do administrador da empresa, o nome de usuário e senha que serão utilizados para autenticação das mensagens trocadas. Estará, automaticamente, habilitado para o processo de testes, período o qual é destinado para adequações no sistema informatizado do cliente. É certo que, nesta fase, todas as informações utilizadas, bem como os documentos emitidos não terão valor fiscal. Detalhe importante é que, todo e-mail enviado pelo web service será encaminhado ao endereço de teste fornecido pelo contribuinte no momento da solicitação do credenciamento.

A conclusão do período de testes deverá ser informada pelo contribuinte e ser comunicada à Receita Municipal para que haja andamento no processo de credenciamento.

Na necessidade, o contribuinte poderá solicitar a troca da senha de acesso ao web service. Durante a fase “Piloto”, o pedido de troca de senha deverá ser encaminhado a partir do e-mail do administrador da empresa para iss@cambe.pr.gov.br. Essa funcionalidade, ao fim da fase “Piloto”, será automatizada via opção “*Configurações*” no menu do usuário administrador do sistema web eNotaFiscal.

Por seu arbítrio, a Receita Municipal poderá solicitar ao contribuinte que forneça a documentação interna na qual basearam-se os testes. Dessa forma, orienta-se o contribuinte a guardar tais informações até que o processo de credenciamento esteja finalizado.

Especificação

O meio de comunicação é a Internet, valendo-se do protocolo HTTPS. As mensagens trocadas entre os sistemas serão envelopes no padrão SOAP com estruturas de dados em XML.

A autenticação do cliente é realizada com o nome de usuário, senha e CNPJ, informados no cabeçalho da mensagem em toda interação que ocorrer. Essas credenciais são fornecidas pela Receita Municipal e obtidas a partir da solicitação de credenciamento.

Os serviços seguem orientação da Associação Brasileira das Secretarias de Finanças das Capitais (ABRASF) e serão disponibilizados gradativamente.

A especificação dos serviços é descrita pelo documento WSDL disponível na seguinte URL:

<https://nfse.cambe.pr.gov.br:8443/enfe-ws/schemas/schema-ws-service.wsdl>

Operações do Web Service

Esta seção apresenta as operações disponíveis no sistema eNotaFiscal-WS no momento da atualização deste manual. Isso significa que, conforme novas operações sejam implantadas no web service, o manual será atualizado.

Cabeçalho das Requisições

Todas as requisições ao web service deverão ser encaminhadas por HTTPS e possuir no cabeçalho a indicação das credenciais do contribuinte: nome de usuário, senha de acesso e CNPJ.

Por exemplo:

```
<soapenv:Header>  
  <auth:authentication xmlns:auth="http://www.cambe.pr.gov.br.com/nfse-ws/security">  
    <auth:username>nomeDeUsuarioWS</auth:username>  
    <auth:password>senhaGeradaPeloSistema</auth:password>  
    <auth:CNPJ>12345678910112</auth:CNPJ>  
  </auth:authentication>  
</soapenv:Header>
```

Geração de NFS-e

Esta funcionalidade tem a função de receber as informações a respeito de uma prestação de serviço e objetiva a emissão de uma NFS-e. Após o processamento da requisição, o sistema remete a resposta contendo alguns dados da NFS-e emitida e/ou alertas ou uma lista com erros. Trata-se de uma operação síncrona. No documento WSDL, é identificada pelo nome *GeracaoNFSeRequest*.

Na resposta, quando a emissão da NFS-e fora bem sucedida, dentre algumas outras informações, tem-se um link onde é possível obter o DANFE em PDF e um campo em formato texto com o conteúdo do espelho em XML. Observa-se que a Prefeitura de Cambé não permite a impressão de documento algum que não seja aquele emitido pelo link de impressão.

O esquema para leitura do espelho em XML da NFS-e encontra-se no seguinte endereço:

https://nfse.cambe.pr.gov.br:8443/enfe/xsd/nfse-cambe-schema-v1_0.xsd

É possível a utilização desta operação para se converter um RPS, seja tipograficamente impresso seja eletrônico. Ressalta-se que o RPS eletrônico só pode ser convertido em NFS-e via web service.

Os códigos requeridos para campos que exigem informação sobre municípios são os presentes na classificação do IBGE. Os códigos requeridos para campos que exigem informação sobre países, valem-se da classificação do Banco Central do Brasil (BACEN).

Consulta de NFS-e

Esta operação tem a função de buscar uma ou mais NFS-e emitidas ou tomadas pelo prestador. No documento WSDL é identificada pelo nome ConsultarNFSeRequest.

Pode-se listar as notas emitidas tanto pelo web service (eNotaFiscal-WS) quanto pelo emissor web (eNotaFiscal). Os filtros para pesquisa baseiam-se na combinação da faixa de número da NFS-e ou faixa de número de RPS associada à faixa de data de emissão ou faixa de competência, isso para serviços prestados. Para serviços tomados, o filtro permite selecionar a faixa de data de emissão ou faixa de competência.

O retorno dessa operação é o mesmo da operação de emissão síncrona. Assim sendo, o esquema de mapeamento do espelho em XML da NFS-e é o mesmo mencionado no subcapítulo anterior.

RPS Eletrônico

Com a concepção deste web service que visa o intercâmbio de informações entre sistemas informatizados para o trato da NFS-e, surgiu a necessidade do RPS eletrônico, cuja função é a mesma do RPS tipograficamente impresso. Difere-se, porém, pelo fato de ser emitido, também, por um sistema informatizado (neste caso, o sistema do contribuinte prestador).

Observação: O RPS eletrônico só pode ser convertido em NFS-e via web service.

É claro que, como regra geral, todo RPS emitido deve ser convertido em NFS-e no prazo legal estipulado em decreto. Outro detalhe é que o contribuinte precisa de autorização da Receita Municipal para emissão de tal documento. Portanto, o contribuinte deve, antecipadamente ao uso do RPS eletrônico, solicitar a AIDF Eletrônica (AIDF-e). Desta forma, a Receita Municipal analisará o pedido que, em vias normais, será deferido e o web service estará apto a aceitar o uso dos RPS autorizados.

Cabe lembrar que a operação *GeracaoNFSeRequest* não obriga que haja uma conversão de RPS para que seja emitida uma NFS-e.

A AIDF-e é solicitada pelo contador por meio do sistema DES, da mesma forma que a AIDF convencional. O detalhe fica a cargo do campo “*Tipo*”, que deverá ser preenchido com a opção “*Eletrônico*”, conforme demonstrado na figura abaixo.



Espécie	Tipo	Número de Vias	Quantidade de RPS	Numeração
R.P.S.	Eletrônico	1	1	1 a 1

Prefeitura de Cambé

Razão Social: PREFEITURA DE CAMBE
CNPJ: 75.732.057/0001-84

Município: CAMBE
UF: PR
CEP: 86181-900

Logradouro: RUA OTTO GAERTNER
Número: 65

Bairro: CENTRO
Complemento:

E-mail Gráfica: ss@cambe.pr.gov.br
Telefone: (43) 3174-2630

Empresa Usuária dos documentos

Inscrição CMC: 000.022.427
CNPJ: 85.260.319/0001-09
Razão social: PESSOA JURÍDICA PRESTAÇÃO DE SERVIÇOS LTDA
Nome Fantasia:
Telefone:
E-mail Empresa: email@provedor.com.br

Finalizar Requisição

Figura 1 - Solicitação de RPS eletrônico via sistema DES.

Secretaria Municipal de Fazenda

O pedido, então, gerará pendência e será analisado pela Receita Municipal. Ao deferir-se, o contribuinte será informado via e-mail e os RPS estarão prontos para uso/conversão.

Mais informações a respeito da AIDF-e e RPS eletrônico podem ser obtidas a partir do capítulo “Solicitação da AIDF” do “MANUAL BÁSICO PARA OPERAÇÕES DO SISTEMA WEB eNotaFiscal” disponível no seguinte endereço:

https://nfse.cambe.pr.gov.br:8443/enfe/manual-operacao/manual_operacao_enotafiscal.pdf

Considerações Sobre o Contrato WSDL

Esta seção tem o objetivo de apresentar algumas considerações a respeito do contrato WSDL. O contrato, por si só, já é suficiente para informar os nomes, tipos de dados, tamanhos mínimo e máximo e obrigatoriedade ou não dos campos. Em alguns casos existe, inclusive, dicionário de dados com a descrição sobre determinados valores possíveis.

Especificidades em alguns formatos de campo

O formato utilizado para informação de data (*type="xs:date"*) será "ano-mês-dia" (máscara *yyyy-MM-dd*).

O tipo de dado numérico decimal e o monetário (*type="pmc:currency"*), cuja base é o tipo decimal, não utilizarão separador de milhar e o separador decimal será o caractere ponto.

Especificidades em alguns conteúdos de campo

O elemento *tomadorCPFCNPJ* (inserido em *xs:complexType name="DadosTomadorInfo"*) é responsável por receber ou o CPF ou o CNPJ do tomador dos serviços quando este for obrigatório (pessoa física identificada ou pessoa jurídica nacional). Quando CPF, deve ter 11 caracteres numéricos e, quando CNPJ, deve ter 14 caracteres numéricos (deve-se completar os caracteres restantes, quando necessário, com zeros à esquerda).

Os elementos *tomadorMunicipio* (*xs:complexType name="DadosTomadorInfo"*) e *municipioPrestacao* (*xs:complexType name="DemaisDadosInfo"*), quando preenchidos, valer-se-ão da classificação de municípios do IBGE (conjunto de sete caracteres numéricos).

Os elementos *tomadorPais* (*xs:complexType name="DadosTomadorInfo"*) e *paisPrestacao* (*xs:complexType name="DemaisDadosInfo"*), quando preenchidos, deverão seguir a categorização de países do BACEN (conjunto de cinco caracteres numéricos, completando-se com zeros à esquerda quando necessário).

O elemento *servicoISS* (*xs:complexType name="DemaisDadosInfo"*) é numérico e, portanto, não se deve informar máscara. Os valores permitidos para esse elemento constam no anexo I da Lei nº 1.723 (31/12/2003) e suas alterações pela Lei Complementar nº 38 (11/10/2017).

Lista de Erros e Alertas

Em todas as respostas do web service, é possível que erros e/ou alertas sejam enviados. Os erros e alertas são identificados por um código alfanumérico. A interpretação deles dar-se-á por um dicionário de dados próprio do Município associado à lista de erros e alertas padronizados pela ABRASF.

Quando uma resposta possui, pelo menos, uma indicação de erro, certamente o comando requisitado não pode ser efetuado. A existência de um ou mais alertas apenas, no entanto, não significa que houve impedimento na execução da tarefa.

Fica explícito que a lista de erros e alertas do Município pode e sofrerá alterações com o passar do tempo devido às implantações do web service, ao passo que torna-se necessário observar esta seção a cada versão do manual que fora publicada.

Lista de Alertas do Município

Os códigos dos alertas do Município iniciam-se por “AL”, conforme tabela abaixo:

CÓDIGO	DESCRIÇÃO	SUGESTÃO
AL001	Item da Lista de Serviço inexistente.	O Item da lista de serviço informado não consta no Cadastro Econômico do contribuinte (CMC/Inscrição Municipal).
AL002	Não há notas na faixa informada.	Modificar os filtros.
AL003	TRANSAÇÃO SEM VALOR FISCAL. CONTRIBUINTE EM FASE DE HOMOLOGAÇÃO.	AO TÉRMINO DESTA FASE DEVE-SE AVISAR A RECEITA MUNICIPAL PARA ATIVAR O MODO PRODUÇÃO.

Lista de Erros do Município

Os códigos dos erros do Município iniciam-se por “L”. A tabela contendo os códigos, descrições e sugestões encontram-se disponíveis em um arquivo separado no seguinte endereço:

https://nfse.cambe.pr.gov.br:8443/enfe/documentos/lista_erros_enfe-ws.xlsx

Exemplos de códigos-fonte

Entende-se que, atualmente, considerando-se o tempo de publicação das especificações WSDL e da disseminação do web service SOAP (a versão 2.0 do WSDL permeia o ano 2005), o consumo de web services não constitui uma completa novidade. Nota-se, também, que é imprescindível existir elemento humano com tal conhecimento em toda equipe de desenvolvimento.

Dessa forma, informa-se que o objetivo deste capítulo é apenas fornecer subsídio básico. Não intenciona-se condicionar tendências, impor padrões ou ensinar o “como” fazer. Obviamente, na maioria das linguagens de programações modernas, existem bibliotecas e/ou componentes mais práticos e eficientes. Sabendo-se disso, observa-se que as informações aqui apresentadas devem ser utilizadas apenas como referência por aqueles que necessitem de um ponto de partida no conceito de web services. Orienta-se, conforme a necessidade, buscar treinamento especializado.

Os exemplos a seguir baseiam-se totalmente no contrato WSDL. Vale lembrar que cada prestador de serviços possui particularidades e os exemplos podem não conter todos os campos necessários para várias situações e, também, conter campos desnecessários para outras.

Ferramenta SoapUI

Em caráter sugestivo, apresenta-se a ferramenta SoapUI (soapui.org), a qual mostra-se de grande valia para entendimento e testes no consumo de web services.

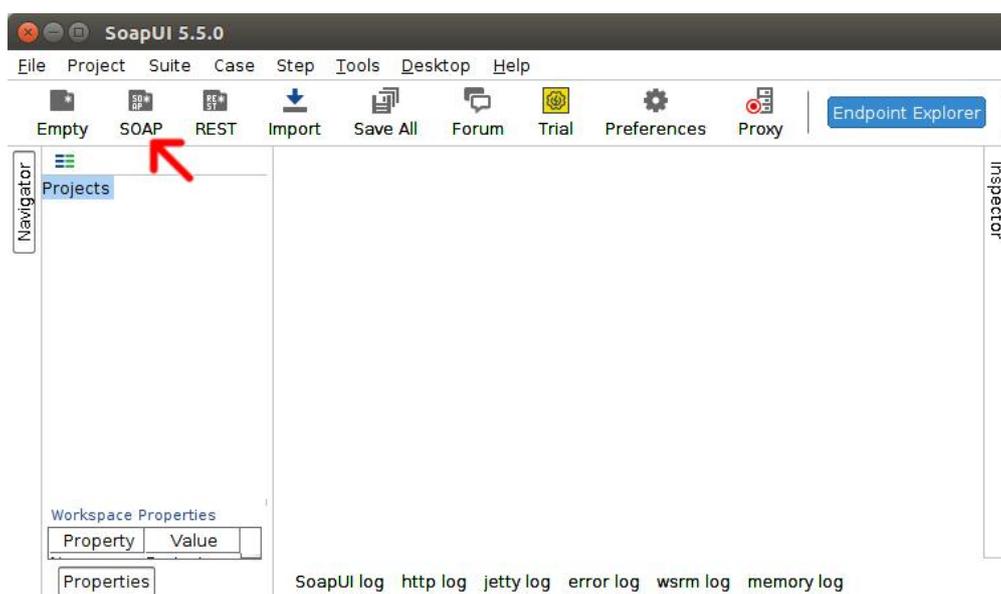


Figura 1 - Botão "SOAP" para se criar um novo projeto.

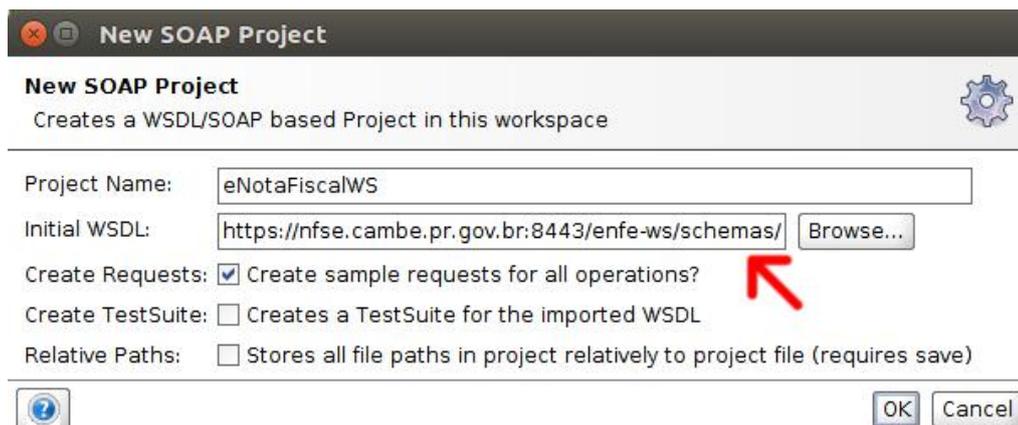


Figura 2 - Insira a URL do contrato WSDL.

O SoapUI comunica-se com o web service, obtém as informações necessárias e monta, de forma automática, uma estrutura inicial. Faz-se necessário adicionar o cabeçalho e preencher os campos desejados. Após, submete-se a requisição. A resposta é apresentada no lado direito da janela da requisição.

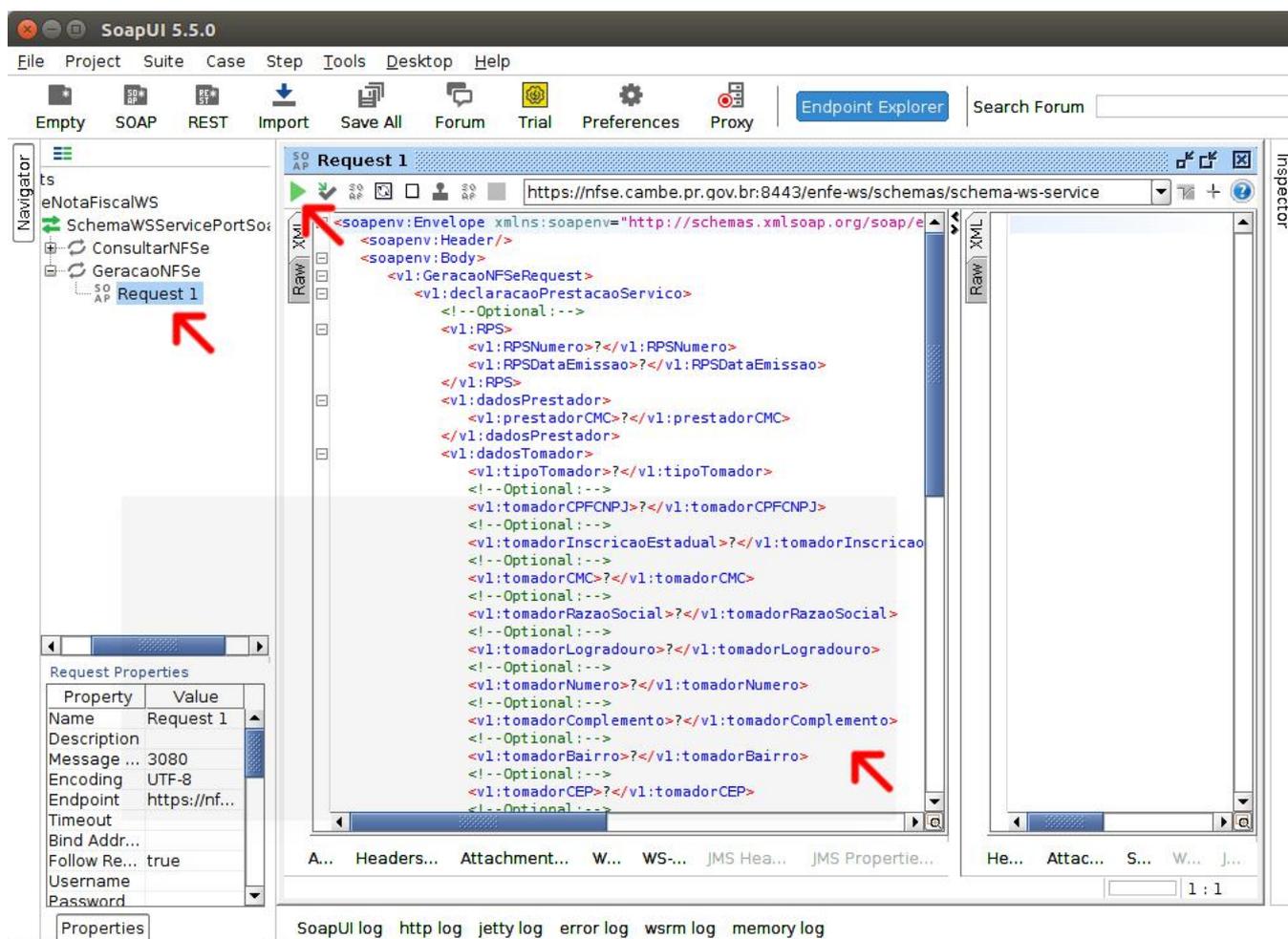


Figura 3 - Preenchimento da requisição e submissão ao web service.

Consumo em JAVA

Valendo-se apenas de classes do pacote *javax.xml*, segue um exemplo de consumo do web service eNotaFiscal-WS utilizando-se da linguagem de programação JAVA.

```
import javax.xml.soap.MessageFactory;
import javax.xml.soap.SOAPBody;
import javax.xml.soap.SOAPConnection;
import javax.xml.soap.SOAPConnectionFactory;
import javax.xml.soap.SOAPElement;
import javax.xml.soap.SOAPEnvelope;
import javax.xml.soap.SOAPHeader;
import javax.xml.soap.SOAPMessage;
import javax.xml.soap.SOAPPart;
import javax.xml.transform.Source;
import javax.xml.transform.Transformer;
import javax.xml.transform.TransformerFactory;
import javax.xml.transform.stream.StreamResult;

public class ExemploRequisicao {

    public static void main(String args[]) {

        Try {
            SOAPConnectionFactory soapConnectionFactory = SOAPConnectionFactory.newInstance();
            SOAPConnection soapConnection = soapConnectionFactory.createConnection();
            String url = "https://nfse.cambe.pr.gov.br:8443/enfe-ws/schemas/schema-ws-service.wsdl";
            SOAPMessage soapRequest = createSoapRequest();

            SOAPMessage soapResponse = soapConnection.call(soapRequest, url);
            createSoapResponse(soapResponse);
            soapConnection.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    private static SOAPMessage createSoapRequest() throws Exception {
        MessageFactory messageFactory = MessageFactory.newInstance();
        SOAPMessage soapMessage = messageFactory.createMessage();
        SOAPPart soapPart = soapMessage.getSOAPPart();

        SOAPEnvelope soapEnvelope = soapPart.getEnvelope();
        soapEnvelope.addNamespaceDeclaration("v1",
            "http://www.cambe.pr.gov.br.com/nfse-ws/xmlschema/v1_0");

        SOAPHeader soapHeader = soapEnvelope.getHeader();
        SOAPElement soapHeaderElement = soapHeader.addChildElement("authentication", "auth",
            "http://www.cambe.pr.gov.br.com/nfse-ws/security");

        SOAPElement soapHeaderUsernameElement = soapHeaderElement.addChildElement("username", "auth");
        soapHeaderUsernameElement.addTextNode("< AQUI VAI O USERNAME >");

        SOAPElement soapHeaderPasswordElement = soapHeaderElement.addChildElement("password", "auth");
        soapHeaderPasswordElement.addTextNode("< AQUI VAI O PASSWORD >");
    }
}
```

```
SOAPElement soapHeaderCNPJElement = soapHeaderElement.addChildElement("CNPJ", "auth");
soapHeaderCNPJElement.addTextNode("< AQUI VAI O CNPJ >");

SOAPBody soapBody = soapEnvelope.getBody();
SOAPElement soapGeracaoNFSeRequestElement = soapBody.
    addChildElement("GeracaoNFSeRequest", "v1");

SOAPElement soapdeclaracaoPrestacaoServicoElement = soapGeracaoNFSeRequestElement.
    addChildElement("declaracaoPrestacaoServico", "v1");

SOAPElement soapDadosPrestador = soapdeclaracaoPrestacaoServicoElement.
    addChildElement("dadosPrestador", "v1");

SOAPElement prestadorCMCElement = soapDadosPrestador.
    addChildElement("prestadorCMC", "v1");
prestadorCMCElement.addTextNode("< AQUI VAI CMC >");

SOAPElement soapDadosTomador = soapdeclaracaoPrestacaoServicoElement.
    addChildElement("dadosTomador", "v1");
SOAPElement soapTipoTomadorElement = soapDadosTomador.
    addChildElement("tipoTomador", "v1");
soapTipoTomadorElement.addTextNode("4");

SOAPElement soapDemaisDadosElement = soapdeclaracaoPrestacaoServicoElement.
    addChildElement("demaisDados", "v1");
SOAPElement servicoISSElement = soapDemaisDadosElement.
    addChildElement("servicoISS", "v1");
servicoISSElement.addTextNode("0000");

SOAPElement soapIncentivoFiscalElement = soapDemaisDadosElement.
    addChildElement("incentivoFiscal", "v1");
soapIncentivoFiscalElement.addTextNode("false");

SOAPElement soapOptanteSimplesNacionalElement = soapDemaisDadosElement.
    addChildElement("optanteSimplesNacional", "v1");
soapOptanteSimplesNacionalElement.addTextNode("false");

SOAPElement soapAliquotaElement = soapDemaisDadosElement.
    addChildElement("aliquota", "v1");
soapAliquotaElement.addTextNode("5");

SOAPElement soapISSDevidoElement = soapDemaisDadosElement.
    addChildElement("ISSDevido", "v1");
soapISSDevidoElement.addTextNode("1");

SOAPElement soapMunicipioPrestacaoElement = soapDemaisDadosElement.
    addChildElement("municipioPrestacao", "v1");
soapMunicipioPrestacaoElement.addTextNode("4103701");

SOAPElement soapServicoValorElement = soapDemaisDadosElement.
    addChildElement("servicoValor", "v1");
soapServicoValorElement.addTextNode("2500");

SOAPElement soapValorBaseCalculoElement = soapDemaisDadosElement.
    addChildElement("valorBaseCalculo", "v1");
soapValorBaseCalculoElement.addTextNode("2500");
```

RECEITA MUNICIPAL

Rua Pará, 264 | Centro | Cambé | PR | CEP 86.181-240 | Fone: (43) 3174-0570

Secretaria Municipal de Fazenda

```
SOAPElement soapTipoDeducacaoElement = soapDemaisDadosElement.  
    addChildElement("tipoDeducacao", "v1");  
soapTipoDeducacaoElement.addTextNode("1");  
  
SOAPElement soapValorDeducacaoElement = soapDemaisDadosElement.  
    addChildElement("valorDeducacao", "v1");  
soapValorDeducacaoElement.addTextNode("0");  
  
SOAPElement soapValorIRElement = soapDemaisDadosElement.  
    addChildElement("valorIR", "v1");  
soapValorIRElement.addTextNode("0");  
  
SOAPElement soapValorPISElement = soapDemaisDadosElement.  
    addChildElement("valorPIS", "v1");  
soapValorPISElement.addTextNode("0");  
  
SOAPElement soapValorCOFINSElement = soapDemaisDadosElement.  
    addChildElement("valorCOFINS", "v1");  
soapValorCOFINSElement.addTextNode("0");  
  
SOAPElement soapValorCSLLElement = soapDemaisDadosElement.  
    addChildElement("valorCSLL", "v1");  
soapValorCSLLElement.addTextNode("0");  
  
SOAPElement soapValorINSSElement = soapDemaisDadosElement.  
    addChildElement("valorINSS", "v1");  
soapValorINSSElement.addTextNode("0");  
  
SOAPElement soapServicoDiscriminacaoElement = soapDemaisDadosElement.  
    addChildElement("servicoDiscriminacao", "v1");  
soapServicoDiscriminacaoElement.addTextNode("DISCRIMINAÇÃO DOS SERVIÇOS NESTA POSIÇÃO.");  
  
soapMessage.saveChanges();  
soapMessage.writeTo(System.out);  
  
return soapMessage;  
}  
  
private static void createSoapResponse(SOAPMessage soapResponse) throws Exception {  
    TransformerFactory transformerFactory = TransformerFactory.newInstance();  
    Transformer transformer = transformerFactory.newTransformer();  
    Source sourceContent = soapResponse.getSOAPPart().getContent();  
  
    StreamResult result = new StreamResult(System.out);  
    transformer.transform(sourceContent, result);  
}  
}
```

Consumo em PHP

O código abaixo foi escrito com base na classe *SoapClient*, fornecida por <https://www.php.net>.

```
<!DOCTYPE html>
<html lang="pt-br">
<head></head>

<body>

<?php
    $argumentos = array(
        "declaracaoPrestacaoServico" => array(
            "dadosPrestador" => array(
                "prestadorCMC" => 999999
            ),
            "dadosTomador" => array(
                "tipoTomador" => 4
            ),
            "demaisDados" => array(
                "servicoISS" => 1722,
                "incentivoFiscal" => false,
                "optanteSimplesNacional" => false,
                "aliquota" => 5,
                "ISSDevido" => 1,
                "municipioPrestacao" => 4103701,
                "servicoValor" => 1500,
                "valorBaseCalculo" => 1500,
                "tipoDeducao" => 1,
                "valorDeducao" => 0,
                "valorIR" => 0,
                "valorPIS" => 0,
                "valorCOFINS" => 0,
                "valorCSLL" => 0,
                "valorINSS" => 0,
                "servicoDiscriminacao" => "DISCRIMINAÇÃO DOS SERVIÇOS NESTA POSIÇÃO."
            )
        )
    );

    $wsdl = "https://nfse.cambe.pr.gov.br:8443/enfe-ws/schemas/schema-ws-service.wsdl";

    $cliente = new SoapClient($wsdl, array('trace'=>1));

    $ns = 'http://www.cambe.pr.gov.br.com/nfse-ws/security';

    $auth = new stdClass();
    $auth->username = new SoapVar('NomeUsuarioWS', XSD_STRING, null, null, 'username', $ns);
    $auth->password = new SoapVar('senha123', XSD_STRING, null, null, 'password', $ns);
    $auth->CNPJ = new SoapVar('00111222000133', XSD_STRING, null, null, 'CNPJ', $ns);
    $authentication = new SoapVar($auth, SOAP_ENC_OBJECT, null, null, 'authentication', $ns);

    $header = new SoapHeader($ns, 'authentication', $authentication, false);
```

```
$cliente->__setSoapHeaders($header);

try {
    $resposta = $cliente->GeracaoNFSe($argumentos);

    echo '<pre>'; print_r($resposta); echo '</pre>';
} catch (Exception $e) {
    echo "<h2>ERRO!</h2>";
    echo $e->getMessage();
}

?>

</body>

</html>
```