**Lista de Arquivos:**

* actions.php
* Auto.NfeSyslog.class.php
* config.php
* database.class.php
* NfeSyslog.class.php
* NfeSyslog\_xml.class.php
* service.php
* service.process.php

**actions.php**

**Auto.NfeSyslog.class.php**

Arquivo php que contém a classe *AutoNFeSyslog*. Classe que que possui funções para automatizar os processos da NFE, é baseada na classe *AutoToolsNFePHP* que vem na pasta de exemplos. Utiliza-se a API do NfePHP com essa classe e conexão com banco de dados. As funções das classes são:

autoTXTtoXML() sem *parâmetros; converte todos os txts para XML na pasta entradas*

autoEnvNFe() *sem parâmetros; envia para sefaz os arquivos que estão na pasta de validados*

autoEnvNFeImproved() *com dois parâmetros; envia para receita e já coloca como autorizada,* rejeitada ou denegada

autoProtNFe() *sem parâmetros; coloca o protocolo das notas que foram enviadas para receita*

autoPrintSend() $fixpdf parametro se salva ou não os pdfs ; imprime as notas na pasta enviadas/autorizadas

autoSignNFe() *sem parâmetros; assina todas as notas na pasta entrada*

autoValidNFe() *sem parâmetros; valida todas as notas na pasta assinadas*

**config.php**

Arquivo que contém a instancia das classes*: DBConn, NfeSyslog\_Service, AutoNFeSyslog.* Já instancia com os parâmetros necessários para trabalhar, interligando alguma delas. Os nomes das instancias(na msm ordem das classes citadas): $db\_nfe, $service, $auto. Ao incluir esse arquivo é possível já trabalhar com a instancias das classes.

Obs: *Para alterar dados como conexão de banco. Caminho do serviço e arquivo php principal do serviço é só alterar nesse arquivo.*

**database.class.php**

Arquivo php que contém a classe *DBConn*. Classe usada para fazer conexão com o banco de dados, utilizando a classe nativa PDO do PHP. O construtor da classe pede os seguintes parâmetros(na mesma ordem citada):

@host, @usuario, @senha, @inicia\_conexão(padrão Sim)

Os métodos da classe são:

connect(); *sem parâmetros*

lookup($sql) *$sql=query a ser executada; retorna o resultado*

exec($sql) *$sql=query a ser executada*

lastInsertId() *sem parâmetros: retorna o ultimo id inserido com o comando exec*

**NfeSyslog.class.php**

**NfeSyslog\_xml.class.php**

Arquivo com a classe *NfeSyslog\_xml*. Classe que faz a ligação do XML com o banco de dados. O construtor da classe pede os seguintes parâmetros(na mesma ordem citada):

@ *DBConn, e @XmlString*

*Os métodos da classe são:*

**Publicas:**

getChave() *sem parâmetros; retorna a chave*

getNumeroNota() *sem parâmetros; retorna o número da nota*

getNaturezaOperacao() *sem parâmetros; retorna a natureza de operação*

getSerieNote() *sem parâmetros; retorna a serie da nota*

getEmitente() *sem parâmetros; retorna o emitente*

SaveNFE() *sem parâmetros; salva a nota no banco(só tem efeito caso seja a primeira vez)*

incAndamento($num) *adiciona andamento na nfe; sem retorno*

saveValidade() *sem parâmetros; salva o andamento de nota validada*

saveEnvio() *sem parâmetros; salva o andamento de nota enviada para sefaz*

saveAprovada() *sem parâmetros; salva o andamento de nota aprovada na sefaz*

saveReprovada($motivo = "") *parâmetro do motivo de rejeição; salva o andamento de nota rejeitada na sefaz*

getRecibo() *sem parâmetros; retorna o recibo*

saveRecibo($recibo) *sem parâmetros; salva o recibo da sefaz*

saveDenegada() *sem parâmetros; salva o andamento de nota denegada*

savePrinting() *sem parâmetros; salva o andamento de danfe impressa*

saveErrorValidade() sem parâmetros;

**Privadas:**

getBdId() *sem parâmetros; retorna o id da nota no banco, só é possivel para uso interno*

InsertBd() *sem parâmetros; insere a nota no banco, só é possivel para uso interno*

getAndamentoCodeNum($code) *parametro do código d andamento; retorna o texto(parecido com lookup manual*

**service.php**

**service.process.php**