

MANUAL DE INTEGRAÇÃO COM O
FORTES FINANCEIRO
(WEB SERVICE)

VERSÃO 2.02

Conteúdo

Histórico de Versões.....	7
Introdução.....	8
Pré-requisito para a Configuração do Ambiente.....	8
Configuração do Ambiente.....	8
Configurações Padrão do Sistema.....	14
Figura 9.....	15
Importando o WSDL no Delphi.....	15
Usando o aplicativo de teste do WebService.....	16
1.Ter um banco de dados do Fortes Financeiro.....	16
2.Configurar o caminho do banco de dados no AGWS.ini.....	16
3.Executar o aplicativo de teste do Web Service.....	17
4.Incluindo informações.....	17
Interfaces disponíveis no WebService – Integração WSDL.....	18
1.Cliente.....	18
1.1. Classe TCliente.....	18
1.2. Classe TServicosCliente.....	21
1.3. Métodos.....	21
2. Fornecedor.....	26
2.1. Classe TFornecedor.....	26
2.2. Métodos.....	28
3. Serviço.....	30
3.1. Classe TServico.....	30
3.2. Métodos.....	31
4. Item.....	32
4.1. Classe TItem.....	32
4.2. Métodos.....	35
5. Lançamentos.....	38
5.1. Classe TLancamentos.....	38
5.2. Classe TRateioLancamentos.....	39
5.3. Métodos.....	39
6. Contas a Receber.....	40
6.1. TContasaReceber.....	40
6.2. TVencimentosaReceber.....	41

6.3. TVencimentosChequeaReceber.....	42
6.4. TVencimentosCartaoaReceber.....	43
6.5. TItensaReceber.....	44
6.5.1. TComissionadosItensaReceber.....	45
6.6. TServiceaReceber.....	45
6.6.1. TComissionadosServiceaReceber.....	46
6.7. TBaixaContasaReceber.....	46
6.8. TBaixaVencimentoCartaoaReceber.....	46
6.9. TRateiosaReceber.....	47
6.10. Métodos.....	47
7. Transferências entre Contas Financeiras.....	57
7.1. TTransferenciaEntreContas.....	57
7.2. Métodos.....	57
8. Adiantamento a clientes.....	58
8.1. TAdiantamentoCliente.....	58
8.2. Métodos.....	59
9. Notas Fiscais de Saída.....	62
9.1. TNotasFiscaisSaida.....	62
9.1.1. TItensNotaFiscalSaida.....	63
9.1.2. TServiceNotaFiscalSaida.....	64
9.2. Métodos.....	65
10. Contas a Pagar.....	67
10.1. TContasaPagar.....	67
10.1.1. TVencimentosaPagar.....	68
10.1.2. TBaixaVencimentoaPagar.....	68
10.1.3. TDespesasaPagar.....	68
10.2. Métodos.....	69
11.1. TAdiantamentoFornecedor.....	71
11.2. Métodos.....	71
12. Requisições.....	72
12.1. TRequisicoes.....	72
12.2. TItensRequisicao.....	73
12.3. Métodos.....	73
13. Entradas.....	74
13.1. TEntradas.....	74

13.2. TItensEntrada.....	75
13.3. Métodos.....	75
14. Saídas.....	76
14.1. TSaídas.....	76
14.2. TItensSaida.....	76
14.3. Métodos.....	77
15. Devoluções de Saída.....	78
15.1. TDevolucoesSaida.....	78
15.2. TItensDevolucoesSaida.....	78
15.3. Métodos.....	79
16. Pedido de Venda	79
16.1. TPedidoDeVenda.....	79
16.2. TItensPedidoDeVenda.....	80
16.3. TServicosPedidoDeVenda.....	81
16.4. Métodos.....	81
17. Receitas e Despesas	82
17.1. TReceitasDespesas.....	82
17.2. Métodos.....	83
18. Centro de Resultados.....	84
18.1. TCentroDeResultado.....	84
18.2. Métodos.....	84
Interfaces disponíveis no Webservice – Integração JSON.....	86
1. Cliente.....	87
1.1. Estrutura do JSON.....	87
1.2. Métodos	88
2. Fornecedor.....	89
2.1. Estrutura do JSON.....	89
2.2. Métodos	90
3. Serviço [Não implementado].....	91
4. Item.....	91
4.1. Estrutura do JSON.....	91
4.2. Métodos.....	91
5. Lançamento.....	92
5.1. Estrutura do JSON.....	92
5.2. Métodos.....	93

6. Contas a Receber.....	94
6.1. Estrutura do JSON	94
6.1.1.Contas a Receber.....	94
6.1.2. Baixa do Contas a Receber.....	96
6.1.3. Baixa do Contas a Receber Cartão.....	97
6.1.4. Baixa do Contas a Receber Cheque.....	97
6.1.5. Cancela Baixa de Vencimentos.....	97
6.2. Métodos.....	98
6.2.1. Contas a Receber.....	98
7. Transferências entre Contas.....	101
7.1. Estrutura do JSON.....	101
7.2. Métodos.....	102
8. Adiantamento a Cliente [Não implementado].....	102
9. Nota Fiscal de Saída [Não implementado].....	102
10. Contas a Pagar.....	103
10.1. Estrutura do JSON.....	103
10.1.1. Contas a Pagar.....	103
10.1.2. Baixas do Contas a Pagar.....	103
10.2. Métodos.....	104
11. Adiantamento a Fornecedor [Não implementado].....	105
12. Requisições [Não implementado].....	105
13. Entradas [Não implementado].....	105
14. Saídas [Não implementado].....	105
15. Devoluções de Saída [Não implementado].....	105
16. Pedido de Venda [Não implementado].....	105
17. Receitas e Despesas.....	105
17.1. Estrutura do JSON.....	105
17.2. Métodos.....	105
Verificando a informação no Fortes Financeiro.....	106
Tabelas Auxiliares.....	107
Tabela 1 – Finalidades DOC.....	107
Tabela 2 - Leiautes FEBRABAN.....	108
Tabela 3 - Tipos de Fatura.....	108
Tabela 4 - Código Situação Tributária coluna A (CSTA).....	108
Tabela 5 - Código Situação Tributária coluna B (CSTB).....	108

Tabela 6 - Código de Situação da Operação no Simples Nacional (CSOSN).....	109
Tabela 7 - Tipo Tributação ISS.....	109
Tabela 8 - Código da Situação Tributária do COFINS/PIS (Tabela 18 do AC Fiscal).....	110
Tabela 9 - Tipo Tributação ICMS/IPI (Tabela 13 do layout do AC Fiscal).....	111
Tabela 10 - Espécie dos Documentos (Tabela 10 do Layout do AC Fiscal).....	111
Tabela 11 - Situação dos Documentos (Tabela 26 do layout do Fiscal).....	112
Tabela 12 - Unidades de Medida (Tabela 15 do layout do AC Fiscal).....	113
Tabela 13 - Tipo de Cálculo do Frete (Tabela 14 do layout do AC Fiscal).....	113
Tabela 14 - Natureza do Frete (Tabela 28 do layout do AC Fiscal).....	113
Tabela 15 - Responsável Frete (Tabela 11 do layout do AC Fiscal).....	113
Tabela 16 - Natureza do Crédito (Tabela 29 do layout do AC Fiscal).....	113
Tabela 17 - Origem do Contas a Pagar.....	114
Tabela 18 - Motivo do Inventário (Tabela 34 do layout do AC Fiscal).....	114
Tabela 19 - Tipo de Tributação do ICMS (Tabela 16 do layout do Fiscal).....	115
Tabela 20 - Situação Tributária do IPI.....	115

Histórico de Versões

2.0 – Reformulação do documento.

Introdução

Este manual tem como finalidade orientar a integração dos sistemas parceiros com Fortes Financeiro através da utilização do nosso *WebService*. O *WebService* é um “espelho” da DLL utilizada anteriormente para a integração. A criação do mesmo foi necessária para aumentar a abrangência deste serviço.

Os arquivos necessários para a integração com o Fortes Financeiro devem ser extraídos do pacote de atualização do sistema.

Pré-requisito para a Configuração do Ambiente

Ter um banco de dados do Fortes Financeiro disponível para acesso, ou seja, um servidor de banco de dados funcionando, onde exista um banco de dados deste sistema. Se o banco de dados do Fortes Financeiro for MS SQL Server, o instalador do Apache deverá ser executado completamente e em seguida o arquivo “AGWS.ini” deverá ser editado manualmente conforme instruções repassadas mais adiante.

Configuração do Ambiente

O primeiro passo para utilização do *WebService* é configurar o servidor de aplicação Apache. Isso pode ser feito através do arquivo “ApacheFortes.exe”. Ao executar este arquivo será feita a instalação e configuração padrão do servidor de aplicação. Caso seja necessário, é possível realizar posteriormente, alterações manuais na configuração.

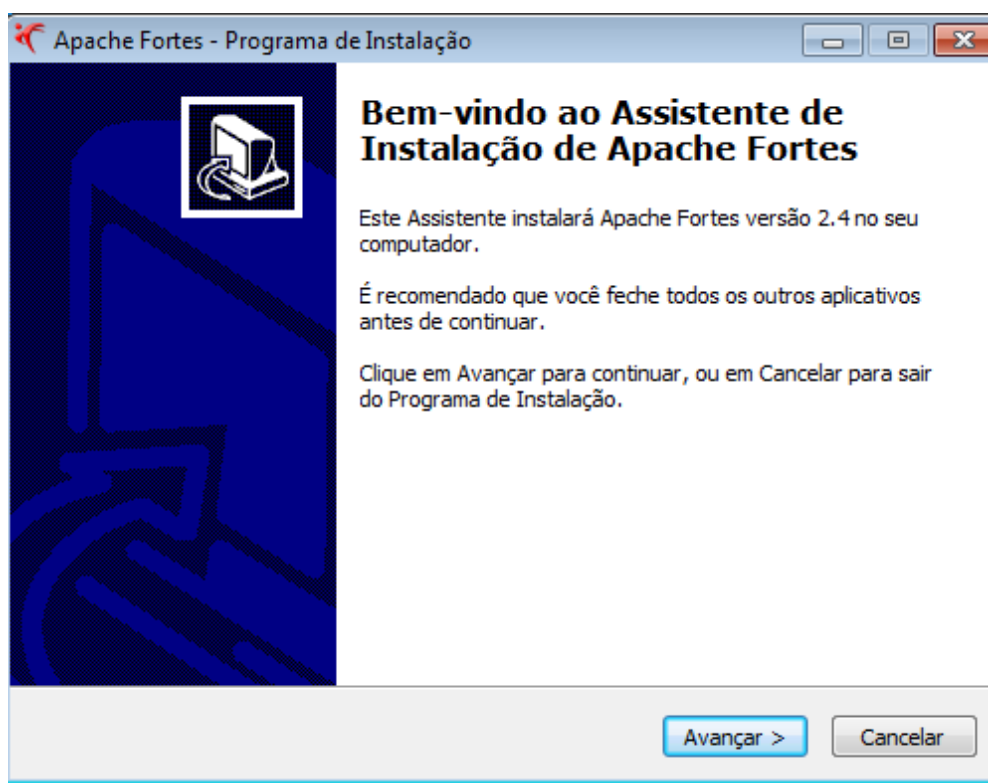
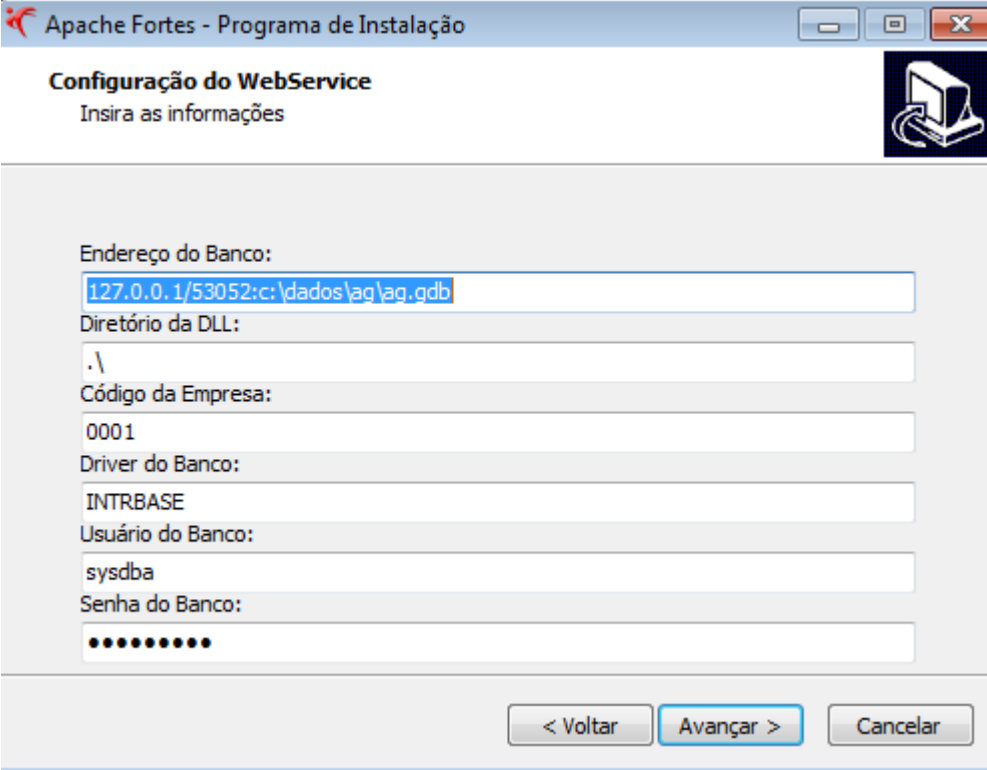


Figura 1

Na tela seguinte [Figura 2] informar os dados solicitados, caso o banco de dados do Fortes Financeiro seja Firebird. Caso o banco de dados seja MS SQL Server, clique em “Avançar” e edite o arquivo “AGWS.ini” ao final da instalação (instruções a seguir).



Apache Fortes - Programa de Instalação

Configuração do Webservice
Insira as informações

Endereço do Banco:
127.0.0.1/53052:c:\dados\ag\ag.qdb

Diretório da DLL:
.\

Código da Empresa:
0001

Driver do Banco:
INTRBASE

Usuário do Banco:
sysdba

Senha do Banco:
●●●●●●●●

< Voltar Avançar > Cancelar

Figura 2

Na tela seguinte [Figura 3] clique o botão “Instalar”.

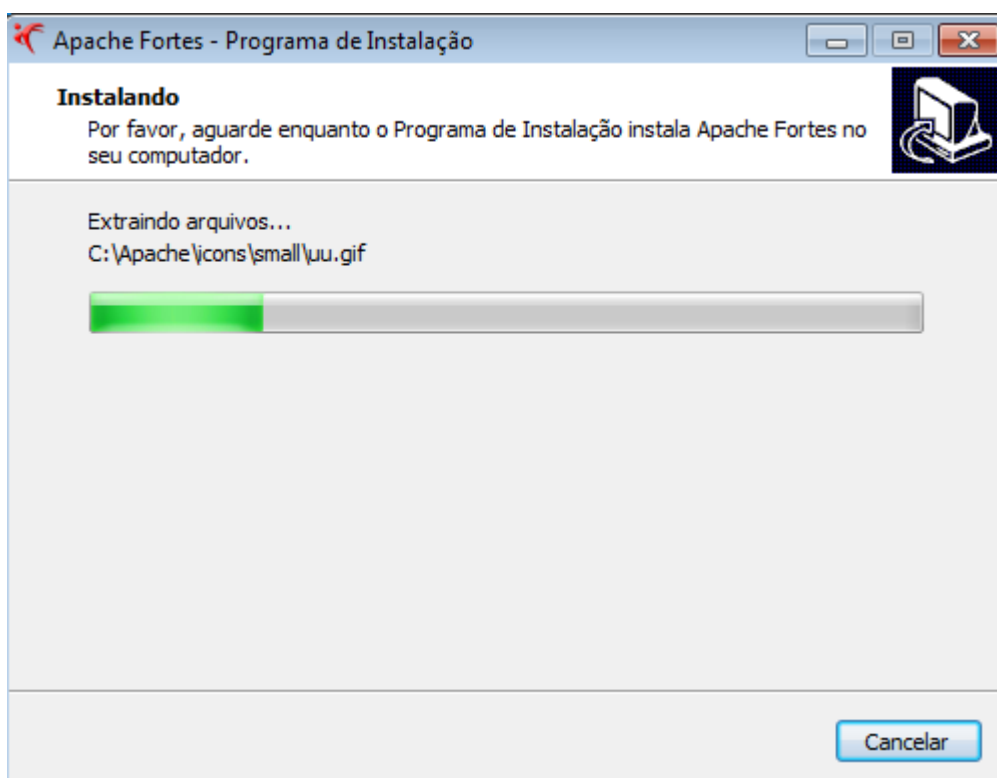
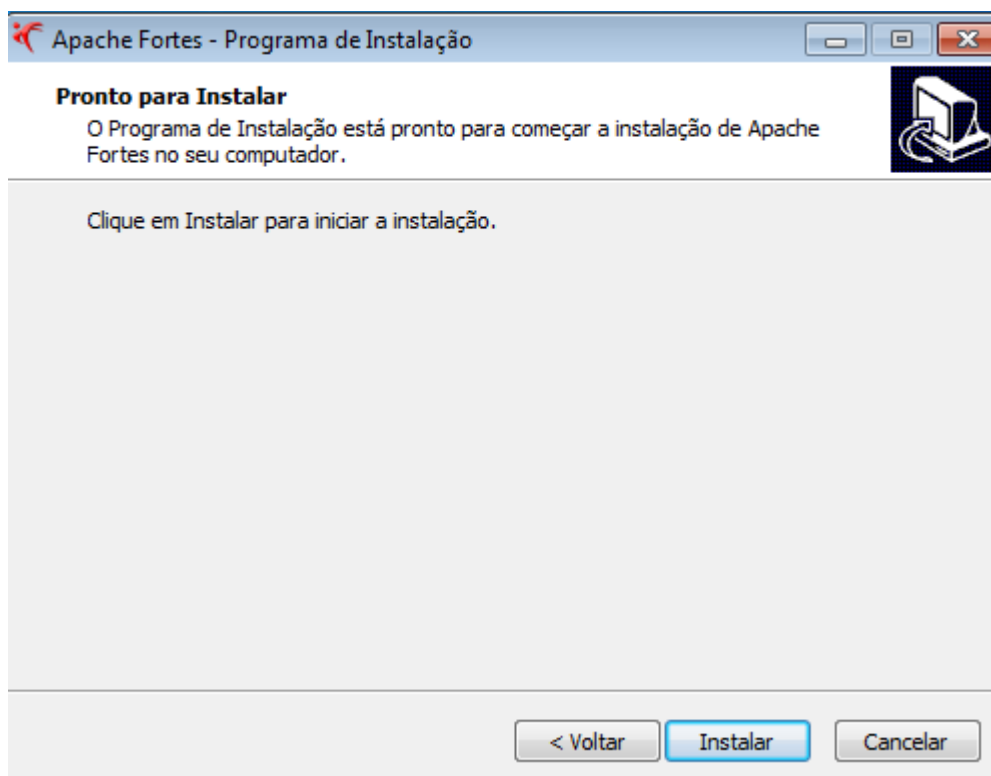


Figura 3

Figura 4

Na tela seguinte [Figura 5] clique o botão “Concluir” para finalizar a instalação.

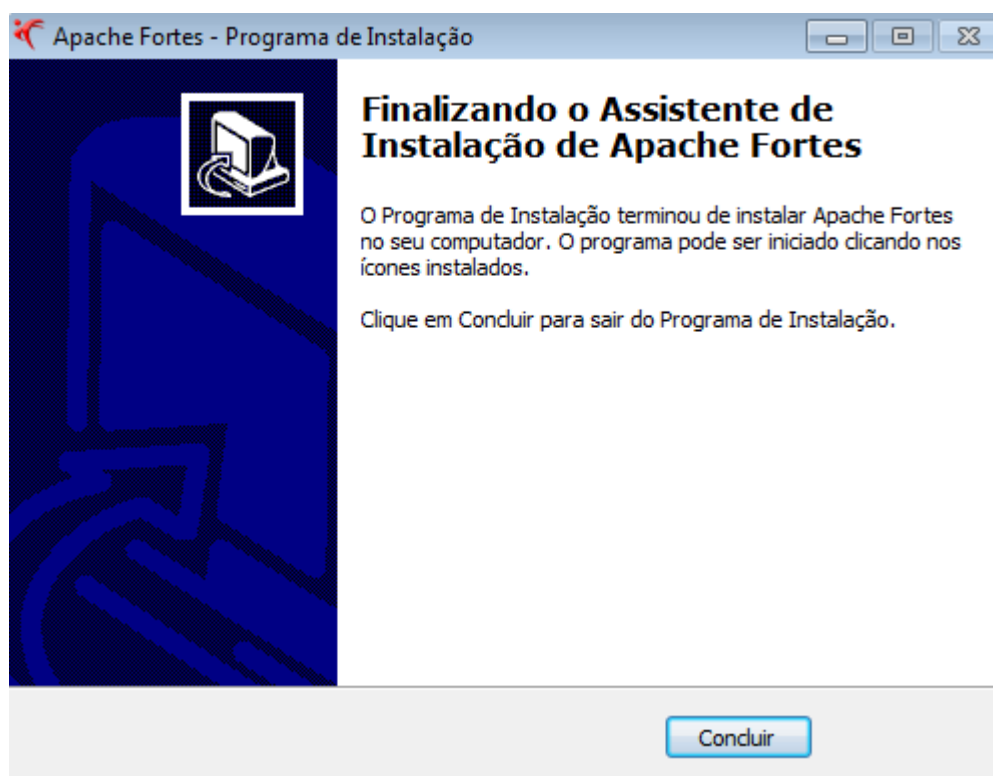


Figura 5

A instalação padrão do servidor Apache cria o diretório: " **C:\{Caminho de pastas} \Apache\cgi-bin**". Na pasta **cgi-bin**, devemos colocar os arquivos: "AGLib.DLL" (dll com as funcionalidades do Fortes Financeiro), "AGWS.exe" (WebService Fortes Financeiro), "AGWS.ini" e "borlndmm.dll". [Figura 6]

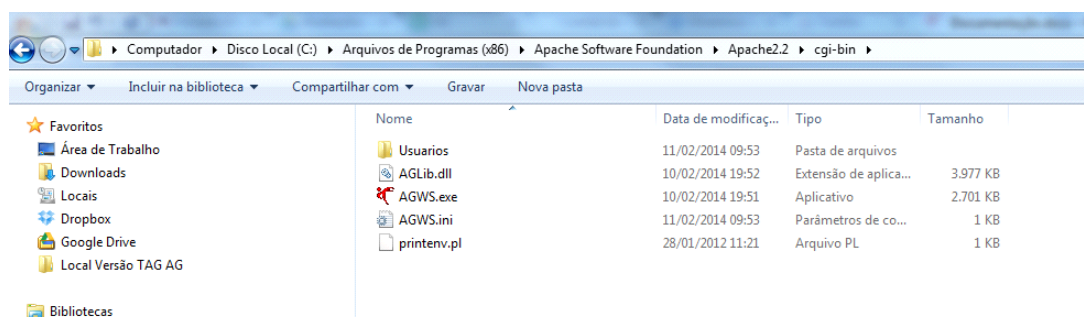


Figura 6

O conteúdo do arquivo "AGWS.ini" deverá conter as seguintes informações:

1. Endereço do Banco = Endereço do banco de dados do Fortes Financeiro.
2. Diretório da DLL = Diretório onde está a DLL do Fortes Financeiro.

3. Driver do Banco = Driver utilizado pelo banco de dados do cliente.
4. Usuário do Banco= Usuário de acesso ao banco de dados.
5. Senha do Banco= Senha de acesso ao banco de dados.
6. Código da Empresa=Código da Empresa da integração.

Exemplo do arquivo “AGWS.ini” quando o banco de dados é Firebird.

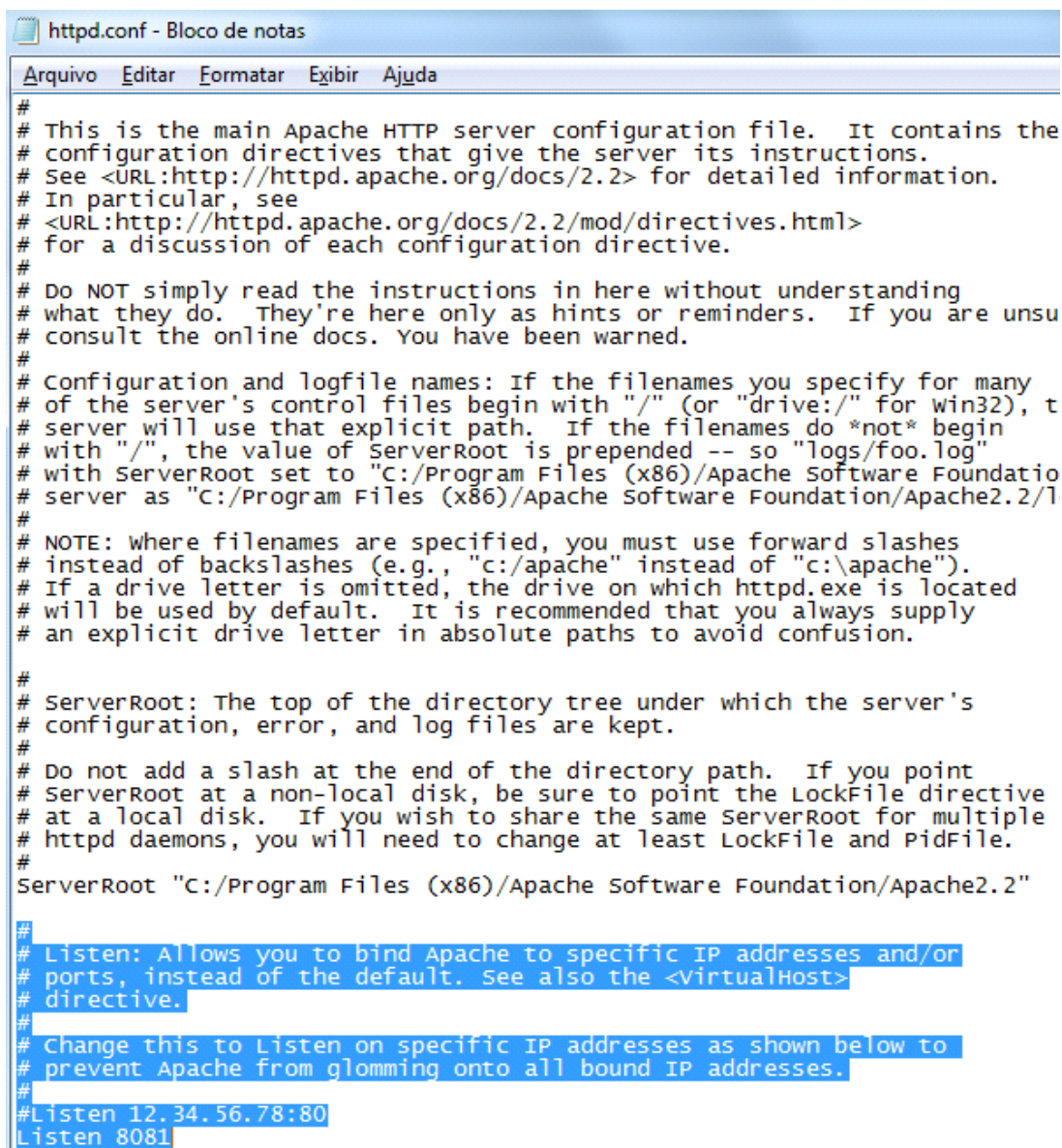
```
[CONFIG]
Endereço do Banco=127.0.0.1/3052:c:\Fortes\ag\ag.gdb
Diretório da DLL=C:\Fortes\ag
Driver do Banco=INTRBASE
Usuário do Banco=sysdba
Senha do Banco=masterkey
Código da Empresa=0001
```

Exemplo do arquivo “AGWS.ini” quando o banco de dados é MS SQL Server.

```
[CONFIG]
Endereço do Banco=127.0.0.1\SQLSERVER2008:AG
Diretório da DLL=C:\Fortes\ag
Driver do Banco=MSSQL
Usuário do Banco=sysdba
Senha do Banco=masterkey
Código da Empresa=0001
```

Para configurar uma porta diferente para acesso ao Apache devemos alterar o arquivo “httpd.conf” da pasta “**C:\{Caminho de pastas}\Apache\conf**”, na seção *Listen* conforme a Figura 7.

Nesse exemplo a porta foi alterada para 8081.



```
#
# This is the main Apache HTTP server configuration file. It contains the
# configuration directives that give the server its instructions.
# See <URL:http://httpd.apache.org/docs/2.2> for detailed information.
# In particular, see
# <URL:http://httpd.apache.org/docs/2.2/mod/directives.html>
# for a discussion of each configuration directive.
#
# Do NOT simply read the instructions in here without understanding
# what they do. They're here only as hints or reminders. If you are unsu
# consult the online docs. You have been warned.
#
# Configuration and logfile names: If the filenames you specify for many
# of the server's control files begin with "/" (or "drive:/" for win32), t
# server will use that explicit path. If the filenames do *not* begin
# with "/", the value of ServerRoot is prepended -- so "logs/foo.log"
# with ServerRoot set to "C:/Program Files (x86)/Apache Software Foundatio
# server as "C:/Program Files (x86)/Apache Software Foundation/Apache2.2/l
#
# NOTE: Where filenames are specified, you must use forward slashes
# instead of backslashes (e.g., "c:/apache" instead of "c:\apache").
# If a drive letter is omitted, the drive on which httpd.exe is located
# will be used by default. It is recommended that you always supply
# an explicit drive letter in absolute paths to avoid confusion.
#
# ServerRoot: The top of the directory tree under which the server's
# configuration, error, and log files are kept.
#
# Do not add a slash at the end of the directory path. If you point
# ServerRoot at a non-local disk, be sure to point the LockFile directive
# at a local disk. If you wish to share the same ServerRoot for multiple
# httpd daemons, you will need to change at least LockFile and PidFile.
#
ServerRoot "C:/Program Files (x86)/Apache Software Foundation/Apache2.2"
#
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
#
# Change this to Listen on specific IP addresses as shown below to
# prevent Apache from glomming onto all bound IP addresses.
#
#Listen 12.34.56.78:80
Listen 8081
```

Figura 7

Para certificar se de que o *WebService* está funcionando corretamente copie o caminho "<http://127.0.0.1:8081/cgi-bin/AGWS.exe/wsdl/IAG>" na barra de endereço de qualquer *browser* e verifique se a resposta está **igual** a apresentada na Figura 8.

IMPORTANTE: A porta que deverá aparecer depois do IP 127.0.0.1 é a que está no arquivo "httpd.conf", então se o aquivo não foi editado, o caminho deverá ser: "<http://127.0.0.1:8080/cgi-bin/AGWS.exe/wsdl/IAG>".

```

<?xml version='1.0' encoding='utf-8'>
<definitions xmlns="http://schemas.xmlsoap.org/wsd/" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:tns="http://tempuri.org/" xmlns:soap="http://schemas.xmlsoap.org/wsd/soap/"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:mime="http://schemas.xmlsoap.org/wsd/mime/" xmlns:n1="urn:AGIntf" name="IAGService" targetNamespace="http://tempuri.org/">
<type>
<xs:schema xmlns="urn:AGIntf" targetNamespace="urn:AGIntf">
<xs:complexType name="TContasReceber">
<xs:sequence>
<xs:element name="estabelecimento" type="xs:string"/>
<xs:element name="centroResultados" type="xs:string"/>
<xs:element name="cliente" type="xs:string"/>
<xs:element name="receita" type="xs:string"/>
<xs:element name="documento" type="xs:string"/>
<xs:element name="tipoGeracao" type="xs:string"/>
<xs:element name="emissao" type="xs:dateTime"/>
<xs:element name="exportacao" type="xs:string"/>
<xs:element name="ISS" type="xs:double"/>
<xs:element name="IRRF" type="xs:double"/>
<xs:element name="IRIS" type="xs:double"/>
<xs:element name="FIS" type="xs:double"/>
<xs:element name="obs" type="xs:string"/>
<xs:element name="mesAno" type="xs:string"/>
<xs:element name="vencimentosReceber" type="n1:ArrayofVencimentosReceber"/>
<xs:element name="vencimentosChequesReceber" type="n1:ArrayofVencimentosChequesReceber"/>
<xs:element name="vencimentosCartaoReceber" type="n1:ArrayofVencimentosCartaoReceber"/>
<xs:element name="servicosReceber" type="n1:ArrayofServicosReceber"/>
<xs:element name="itensReceber" type="n1:ArrayofItensReceber"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="TArrayofVencimentosReceber">
<xs:complexContent>
<xs:restriction base="soapenc:Array">
<xs:sequence/>
<xs:attribute xmlns:n1="http://schemas.xmlsoap.org/wsd/" ref="soapenc:ArrayType" n1:arrayType="n1:TVencimentosReceber[]"/>
</xs:restriction>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="TArrayofVencimentosChequesReceber">
<xs:complexContent>
<xs:restriction base="soapenc:Array">
<xs:sequence/>
<xs:attribute xmlns:n1="http://schemas.xmlsoap.org/wsd/" ref="soapenc:ArrayType" n1:arrayType="n1:TVencimentosChequesReceber[]"/>
</xs:restriction>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="TArrayofVencimentosCartaoReceber">
<xs:complexContent>
<xs:restriction base="soapenc:Array">
<xs:sequence/>
<xs:attribute xmlns:n1="http://schemas.xmlsoap.org/wsd/" ref="soapenc:ArrayType" n1:arrayType="n1:TVencimentosCartaoReceber[]"/>
</xs:restriction>
</xs:complexContent>
</xs:complexType>
</xs:complexType>
</type>
</definitions>

```

Figura 8

Configurações Padrão do Sistema

Algumas movimentações do *webservice* possuem campos obrigatórios que não são de conhecimento do sistema que está utilizando a integração, como por exemplo: Grupo de Clientes, Receita do Cliente, Grupo de Fornecedores, Despesa do Fornecedor, etc.

Para estes casos, foram criadas configurações padrão no sistema Fortes Financeiro, que serão utilizadas automaticamente nas integrações.

Estas configurações encontram-se em: Utilitários - Configuração - Empresa - Financeiro – Integração - Webservice e DLL.

Lembrando que para utilizar estas configurações é necessário incluir os registros em seus respectivos cadastros.

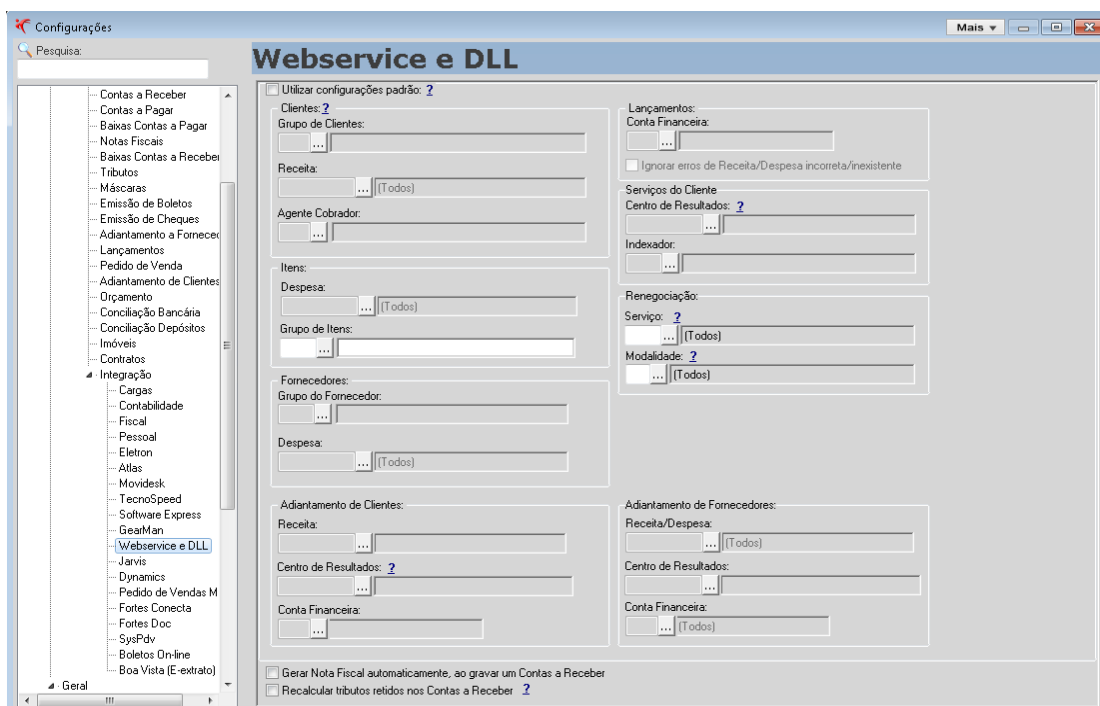


Figura 9

Importando o WSDL no Delphi

Se o sistema que irá integrar com o Fortes Financeiro for desenvolvido em Delphi, existe a opção de importar o WSDL do *WebService*. Ao importar o WSDL será gerado o método “GetIAG” [Figura 10] que deve ser consumido para obter a interface.

```
function GetIAG(UseWSDL: Boolean; Addr: string; HTTPRIO: THHTTPRIO): IAG;
const
  defWSDL = 'http://localhost:8080/cgi-bin/agws.exe/wsdl/IAG';
  defURL = 'http://localhost:8080/cgi-bin/agws.exe/soap/IAG';
  defSvc = 'IAGService';
  defPrt = 'IAGPort';
var
  RIO: THHTTPRIO;
begin
  Result := nil;
  if (Addr = '') then
  begin
    if UseWSDL then
      Addr := defWSDL
    else
      Addr := defURL;
  end;
  if HTTPRIO = nil then
  begin
    RIO := THHTTPRIO.Create(nil)
  end
  else
    RIO := HTTPRIO;
  try
    Result := (RIO as IAG);
    if UseWSDL then
    begin
      RIO.WSDLLocation := Addr;
      RIO.Service := defSvc;
      RIO.Port := defPrt;
    end
    else
      RIO.URL := Addr;
  finally
    if (Result = nil) and (HTTPRIO = nil) then
      RIO.Free;
  end;
end;
```

Figura 10

O consumo do método “GetIAG” pode ser feito conforme mostrado abaixo [Figura 11].

```
function TfrmTeste.GetIAG: IAG;  
begin  
    Result := IAG1.GetIAG(False, edtEnderecoWS.Text, HTTPRIO1);  
end;
```

Figura 11

Após consumir o método “GetIAG”, podemos acessar os métodos da interface. [Figura 12]

```
procedure TfrmTeste.btnGetJSONCLIClick(Sender: TObject);  
begin  
    MemoCLI.Text := GetIAG.GetJsonClientePorCodigo('000001');  
end;
```

Figura 12

Usando o aplicativo de teste do *WebService*

Antes de iniciar a implementação da integração, sugere-se utilizar o aplicativo de testes do *Web Service* (AGWSTeste.exe) para se certificar de que tudo está corretamente instalado e configurado. Este aplicativo fica disponível na pasta " *C:\{Caminho de pastas} \Apache\cgi-bin*", criada no momento em que o "ApacheFortes.exe" é instalado. Para tanto é necessário seguir os passos a seguir.

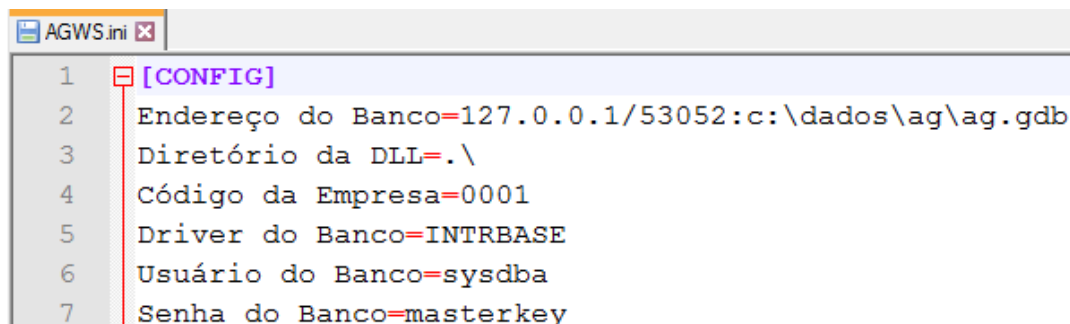
1.Ter um banco de dados do Fortes Financeiro

Na seção “Pré-requisito para a Configuração do Ambiente” falou-se sobre isso. Caso não exista um banco de dados do Fortes Financeiro disponível, será necessário instalar a versão demonstração do sistema (Solicitar o instalador à Fortes Tecnologia). Neste caso o banco de dados firebird(AG.fdb) é instalado no caminho “C:\Fortes\Ag”.

2.Configurar o caminho do banco de dados no AGWS.ini

Após localizar o banco de dados que se deseja usar nos testes do *WebService* é necessário informá-lo no

aquivo AGWS.ini está em: "C:\Apache\cgi-bin" conforme exemplo da [Figura13].



```
1 [CONFIG]
2 Endereço do Banco=127.0.0.1/53052:c:\dados\ag\ag.gdb
3 Diretório da DLL=.\
4 Código da Empresa=0001
5 Driver do Banco=INTRBASE
6 Usuário do Banco=sysdba
7 Senha do Banco=masterkey
```

Figura 13

3.Executar o aplicativo de teste do Web Service

Depois de realizar os passos anteriores, agora já é possível executar o aplicativo de teste (AGWSTeste.exe)do Web Service. O aplicativo está na pasta "C:\Apache\cgi-bin". O primeiro teste que deve ser feito é o teste de ECO, para garantir que a comunicação com o web service está OK. Abaixo [figura 14] é exibida a interface do aplicativo de teste do web service.

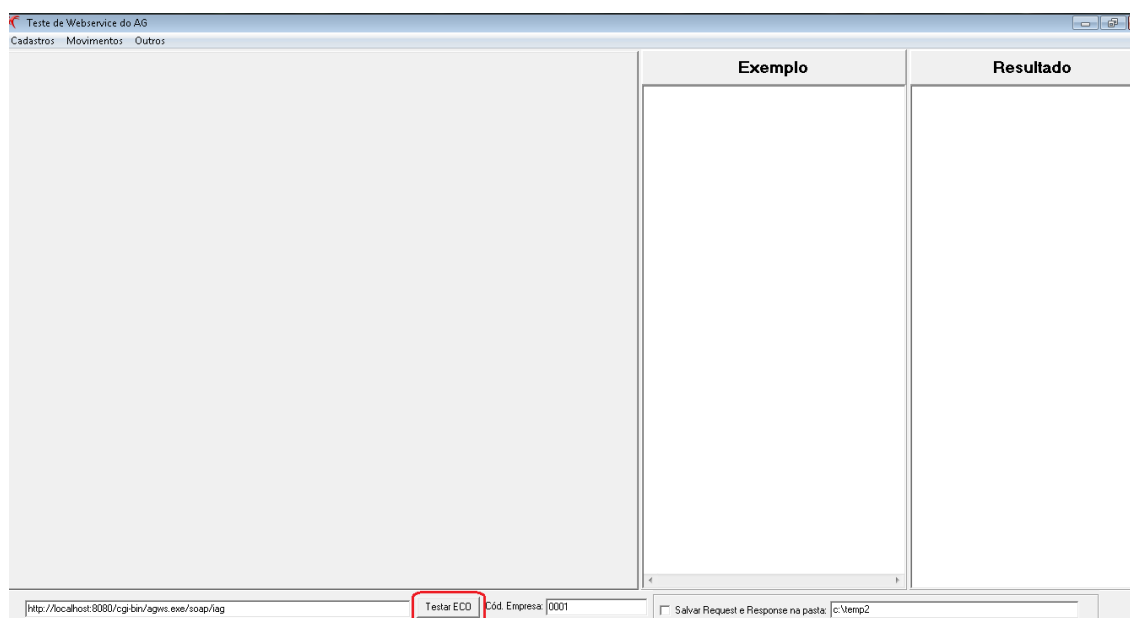


Figura 14

Antes de testar o ECO deve-se conferir se a url informada na parte inferior esquerda do aplicativo está correta. Ao clicar o

botão "Testar ECO" o sistema tentará acessar o web service e exibirá o resultado [Figura 15].

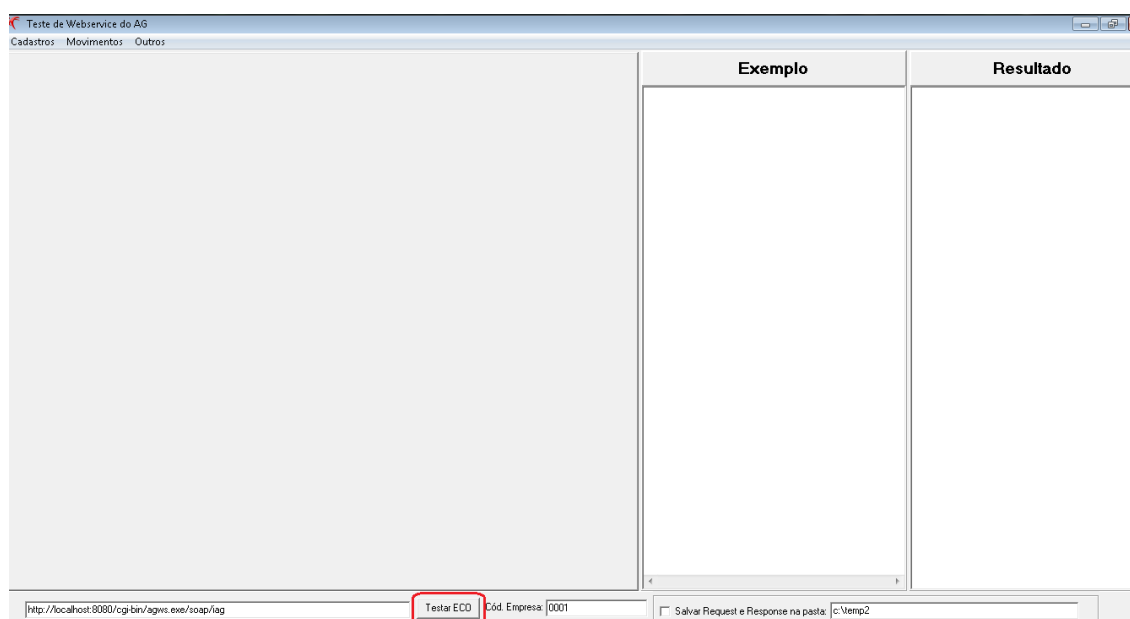


Figura 15

4.Incluindo o informações

Através do aplicativo de testes é possível

incluir/alterar/excluir informações.

No exemplo abaixo, ao clicar o botão “Inclusão Cliente” é consumido o método:
`function IncluiCliente(const Cliente: TCliente): string;`

Este método recebe como parâmetro um objeto do tipo TCliente. Esta classe encontra-se descrita na seção específica. Ao verificar a classe atentar para as dependências, exemplo: para incluir cliente é necessário ter cadastrado no Fortes Financeiro: o grupo do cliente, agente cobrador e a conta de receita/despesa.

Caso o cliente seja incluído com sucesso o *Web Service* retorna o código do cliente cadastrado [Figura 16]. Essa inclusão pode ser consultada via sistema ou banco de dados. Na seção “Verificando a informação no Fortes Financeiro” é possível saber como essa conferência pode ser feita.

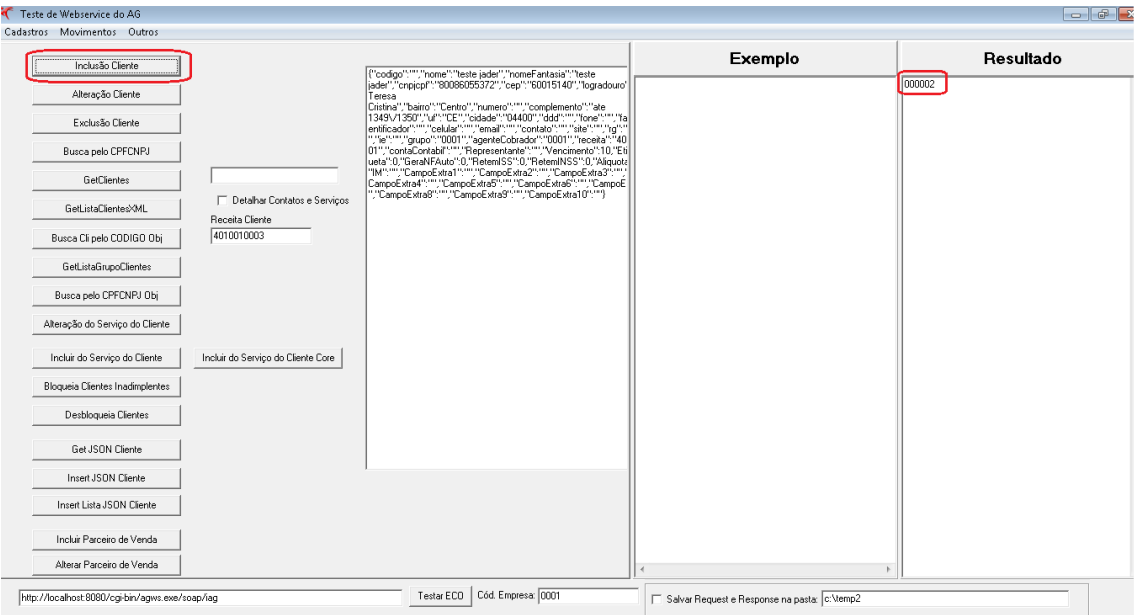


Figura 16

Assim como foi feito para cliente, também é possível

incluir/alterar/excluir outras informações, basta explorar os menus “Cadastros” e “Movimentos” do aplicativo.

Interfaces disponíveis no *WebService* – Integração WSDL

1.Cliente

1.1. Classe TCliente

Campo	Tipo	Descrição	Obrigatório	Observação
Codigo	String(6)	Código do cliente	SIM	Se o campo código não for informado, o <i>webservice</i> vai sugerir o código. Se for informado um tamanho menor do que 6, o <i>webservice</i> completará o código com zeros à esquerda.
Nome	String(80)	Nome do cliente	SIM	

NomeFantasia	String(30)	Nome fantasia da empresa cliente	NÃO	
CNPJCPF	String(14)	CPF/CNPJ do cliente	SIM	
CEP	String(8)	Código Postal do endereço do cliente	NÃO	
Logradouro	String(60)	Nome do logradouro do cliente	NÃO	
Bairro	String(30)	Nome do bairro do cliente	NÃO	
Numero	String(10)	Número do endereço	NÃO	
Complemento	String(20)	Complemento do endereço	NÃO	
UF	String(2)	Unidade Federativa do cliente	NÃO	
Cidade	String(5)	Código IBGE do Município	NÃO	Código IBGE do Município, sem a parte da UF.
DDD	String(4)	Código DDD	NÃO	
Fone	String(20)	Número de telefone	NÃO	
DDD do Fax	String(4)	Código DDD do Fax	NÃO	
Fax	String(10)	Número do Fax	NÃO	
Identificador	String(20)	Código identificador do cliente	NÃO	
DDD do Celular	String(4)	Código DDD do Celular	NÃO	
Celular	String(9)	Número do telefone celular	NÃO	
Email	String(254)	Endereço de e-mail	NÃO	
Contato	String(30)	Nome do contato do cliente	NÃO	
Site	String(60)	Endereço web do cliente	NÃO	
RG	String(20)	Número do RG	NÃO	
IE	String(20)	Inscrição Estadual	NÃO	Inscrição Estadual
Grupo	String(4)	Código do grupo do cliente	SIM	Chave do Grupo de Clientes no Financeiro
AgenteCobrador	String(4)	Código do agente cobrador	SIM	Chave do Agente Cobrador no Financeiro
Receita	String(10)	Código da Receita	SIM	Chave da Receita no Financeiro
ContaContabil	String(15)	Código da Conta Contábil	NÃO	Conta Contábil no Financeiro
Representante	String(4)	Código do representante	NÃO	
Vencimento	Integer	Dia do vencimento	NÃO	Dia para vencimento

				da fatura
Etiqueta	Integer	Indica se o relatório de impressão de etiquetas conterá os dados deste cliente: 0 - NÃO 1 - SIM	NÃO	Utilizado no relatório de geração de Etiquetas.
GeraNFAuto	Integer	Indica se deve ou não gerar Nota Fiscal automaticamente: 0 - NÃO 1- SIM	NÃO	Utilizado na geração do faturamento automático para decidir se vai ou não gerar a Nota Fiscal.
Retem ISS	Integer	Indica se há ou não retenção de ISS nas notas fiscais do cliente: 0 - NÃO 1- SIM	NÃO	Informar se há retenção de ISS
Alíquota ISS	Double	Percentual da alíquota de ISS	NÃO	Alíquota de ISS
Retem INSS	Integer	Indica se há ou não retenção de INSS nas notas fiscais do cliente: 0 - NÃO 1- SIM	SIM	Retenção de INSS
Centro de Resultado	String(10)		NÃO	Chave do Centro de Resultados no Financeiro
IM	String(20)	Inscrição Municipal	NÃO	Inscrição Municipal
CampoExtra1	String(254)		NÃO	
CampoExtra2	String(254)		NÃO	
CampoExtra3	String(254)		NÃO	
CampoExtra4	String(254)		NÃO	
CampoExtra5	String(254)		NÃO	
CampoExtra6	String(254)		NÃO	
CampoExtra7	String(254)		NÃO	
CampoExtra8	String(254)		NÃO	
CampoExtra9	String(254)		NÃO	
CampoExtra10	String(254)		NÃO	
IDWS	String(30)	Identificador do cliente incluído via <i>WebService</i>	NÃO	
É Cliente Pagador	Integer		NÃO	
Cliente Pagador	String(6)		NÃO	
Exterior	Integer	Indica se o cliente é estrangeiro: 0 - NÃO 1- SIM	SIM	

Inseto de IE	Integer	Indica se o cliente é isento de inscrição estadual: 0 - NÃO 1- SIM	SIM	
Indicador de IE Destinatário	Integer		NÃO	
Documento estrangeiro	String(50)		NÃO	
Grupo Econômico	String(4)		NÃO	Chave do Grupo Econômico no Financeiro
ServicosCliente	TServicosCliente		NÃO	

1.2. Classe TServicosCliente

Campo	Tipo	Descrição	Obrigatório	Observação
Servico	String(4)		SIM	
Sequencial	Integer		SIM	
Estabelecimento	String(4)		SIM	
Modalidade	String(2)		SIM	
Indexador	String(4)		SIM	
QtdIndexador	Double		SIM	
AnoMesPrimeiraCobranca	String(6)		SIM	
Aquisicao	XSDatetime		NÃO	
AnoMesReajuste	String(6)		NÃO	
Observacao	Blob		NÃO	
DescricaoNota	String(60)		NÃO	
ComplementoDescricaoNota	Blob		NÃO	
Contrato	String(20)		NÃO	
DtCancelamento	XSDatetime		NÃO	
ObsCancelamento	Blob		NÃO	
CRS_Codigo	String(1)		NÃO	
DescPrevisto	Double		NÃO	

1.3. Métodos

- **IncluiCliente**
 - Assinatura:
`function IncluiCliente(const Cliente: TCliente): string; stdcall;`
 - Parâmetro(s):

- Objeto do tipo TCliente;
- Possíveis retornos:
 - Código do Cliente, caso a inclusão seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a inclusão;
- **AlterarCliente**
 - Assinatura:


```
function AlterarCliente(const Cliente: TCliente): string;
stdcall;
```
 - Parâmetro(s):
 - Objeto do tipo TCliente;
 - Possíveis retornos:
 - Código do Cliente, caso a alteração seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a alteração;
- **ExcluirCliente**
 - Assinatura:


```
function ExcluirCliente(const Cliente: TCliente): string;
overload; stdcall;
```
 - Parâmetro(s):
 - Objeto do tipo TCliente;
 - Possíveis retornos:
 - 'OK', caso a exclusão seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a exclusão;
- **ExcluirClienteWithCNPJCPF**
 - Assinatura:


```
function ExcluirClienteWithCNPJCPF(const CNPJCPF: String):
String; overload; stdcall;
```
 - Parâmetro(s):
 - Objeto do tipo TCliente;
 - Possíveis retornos:
 - 'OK', caso a exclusão seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a exclusão;
- **GetClienteWithIdentificador**
 - Assinatura:


```
function GetClienteWithIdentificador(const Identificador:
String): String; stdcall;
```
 - Parâmetro(s):
 - Identificador (campo identificador passado no momento da inclusão ou alteração do cliente);
 - Possíveis retornos:
 - Código do Cliente, caso exista cliente com o identificador passado;
 - String vazia, caso não exista cliente com o identificador passado;
 - Descrição do Erro, caso aconteça algum erro durante a pesquisa;
- **GetObjetoClienteCPFCNPJ**

- Assinatura:


```
function GetObjetoClienteCPFCNPJ(const CPFCNPJ: string):
TCliente; stdcall;
```
- Parâmetros:
 - CPF/CNPJ do cliente
- Possíveis retornos:
 - Objeto do tipo TCliente, caso exista cliente com o CPF/CNPJ passado;
 - *nil*, caso não exista cliente com o CPF/CNPJ passado;
 - Descrição do Erro, caso aconteça algum erro durante a pesquisa;
- **GetObjetoClienteCodigo**
 - Assinatura:


```
function GetObjetoClienteCodigo(const CODIGO: string):
TCliente; stdcall;
```
 - Parâmetros:
 - Código do cliente;
 - Possíveis retornos:
 - Objeto do tipo TCliente, caso exista cliente com o código passado;
 - *nil*, caso não exista cliente com o código passado;
 - Descrição do Erro, caso aconteça algum erro durante a pesquisa;
- **GetObjetoClienteIDWS**
 - Assinatura:


```
function GetObjetoClienteIDWS(const IDWS: string): TCliente;
stdcall;
```
 - Parâmetros:
 - IDWS (campo utilizado pelo sistema parceiro para cadastrar cliente, caso o mesmo deseje assim);
 - Possíveis retornos:
 - Objeto do tipo TCliente, caso exista cliente com o IDWS passado;
 - *nil*, caso não exista cliente com o IDWS passado;
 - Descrição do Erro, caso aconteça algum erro durante a pesquisa;
- **GetClienteWithCPFCNPJ**
 - Assinatura:


```
function GetClienteWithCPFCNPJ(const CNPJCPF: String): String;
stdcall;
```
 - Parâmetros:
 - CPF/CNPJ
 - Possíveis retornos:
 - Código do Cliente, caso exista cliente com o CPF/CNPJ passado;
 - String em branco, caso não exista cliente com o CPF/CNPJ passado;
 - Descrição do Erro, caso aconteça algum erro durante a busca;
- **GetClienteWithIDWS**
 - Assinatura:


```
function GetClienteWithIDWS(const IDWS: String): String;
stdcall;
```

- Parâmetros:
 - IDWS
- Possíveis retornos:
 - Código do Cliente, caso exista cliente com o IDWS passado;
 - Descrição do Erro, caso aconteça algum erro durante a busca;
- **GetClienteWithIDWSRetornoVazio**
 - Assinatura:


```
function GetClienteWithIDWSRetornoVazio(const IDWS: String): String; stdcall;
```
 - Parâmetros:
 - IDWS
 - Possíveis retornos:
 - Código do Cliente, caso exista cliente com o IDWS passado;
 - String em branco, caso não exista cliente com o IDWS passado;
 - Descrição do Erro, caso aconteça algum erro durante a busca;
- **GetCNPJCPFClienteWithIDWS**
 - Assinatura:


```
function GetCNPJCPFClienteWithIDWS(const IDWS: String): String; stdcall;
```
 - Parâmetros:
 - IDWS
 - Possíveis retornos:
 - CPF/CNPJ do Cliente, caso exista cliente com o IDWS passado;
 - Descrição do Erro, caso aconteça algum erro durante a busca;
- **GetObjetoClienteWithCPFCNPJ**
 - Assinatura:


```
function GetObjetoClienteWithCPFCNPJ(const CNPJCPF: String): TCliente; stdcall;
```
 - Parâmetros:
 - CPF/CNPJ;
 - Possíveis retornos:
 - Objeto do tipo TCliente, caso exista cliente com o CPF/CNPJ passado;
 - *nil*, caso não exista cliente com o CPF/CNPJ passado;
 - Descrição do Erro, caso aconteça algum erro durante a pesquisa;
- **GetListaClientesXML**
 - Assinatura:


```
function GetListaClientesXML(ConsiderarCancelados: Boolean = True; CPFCNPJ: string = ''; DetalhaContatosEServicos: Boolean = False): String; stdcall;
```
 - Parâmetros:
 - Considerar Cancelados;
 - CPFCNPJ;
 - DetalhaContatosEServicos;
 - Possíveis retornos:

- string no formato xml com a lista de clientes selecionados de acordo com os parâmetros passados;
- **GetListaClientes**
 - Assinatura:


```
function GetListaClientes(ConsiderarCancelados: Boolean = True; CPFCNPJ: string = ''; DetalhaContatosEServicos: Boolean = False): String; stdcall;
```
 - Parâmetros:
 - Considerar Cancelados;
 - CPFCNPJ;
 - DetalhaContatosEServicos;
 - Possíveis retornos:
 - string no formato JSON com a lista de clientes selecionados de acordo com os parâmetros passados;
- **GetListaClientesPorEmpresa**
 - Assinatura:


```
function GetListaClientesPorEmpresa(ConsiderarCancelados: Boolean = True; CPFCNPJ: string = ''; DetalhaContatosEServicos: Boolean = False; Emp_Codigo: string = ''): String; stdcall;
```
 - Parâmetros:
 - Considerar Cancelados;
 - CPFCNPJ;
 - DetalhaContatosEServicos;
 - Código da Empresa;
 - Possíveis retornos:
 - string no formato JSON com a lista de clientes selecionados de acordo com os parâmetros passados;
- **IncluiServicosCliente**
 - Assinatura:


```
function IncluiServicosCliente(const CNPJ: String; const ServicosCliente: TServicosCliente): String; stdcall;
```
 - Parâmetro(s):
 - CNPJ;
 - Objeto do tipo TServicosCliente;
 - Possíveis retornos:
 - Sequencial do serviço do cliente, caso a inclusão seja bem sucedida;
 - 'Não foi possível alterar serviço do cliente. (Não foi passado um tipo válido para a função).', caso não seja possível carregar o cliente passado ;
 - 'Não conseguiu acessar o parâmetro repassado para a função: Cliente.CNPJCPF.', caso não consiga acessar os serviços do cliente passado;
 - 'Não foi possível localizar o cliente.', caso o cliente passado não seja encontrado;
 - Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **AlteraServicosCliente**

- Assinatura:

```
function AlteraServicosCliente(const CNPJ:String; const
ServicosCliente: TServicosCliente): String; stdcall;
```
- Parâmetro(s):
 - CNPJ;
 - Objeto do tipo TServicosCliente;
- Possíveis retornos:
 - Código do serviço do cliente, caso a alteração seja bem sucedida;
 - 'Não foi possível alterar serviço do cliente. (Não foi passado um tipo válido para a função).', caso não seja possível carregar o cliente passado ;
 - 'Não conseguiu acessar o parâmetro repassado para a função: Cliente.CNPJCPF.', caso não consiga acessar os serviços do cliente passado;
 - 'Cliente não encontrado.(AlteraServicosCliente)', caso o cliente passado não seja encontrado;
 - 'Serviço do cliente não encontrado.', caso o cliente passado não seja encontrado;
 - Descrição do Erro, caso aconteça algum erro durante a alteração;

2. Fornecedor

2.1. Classe TFornecedor

Campo	Tipo	Descrição	Obrigatório	Observação
Codigo	String(6)	Código do fornecedor	SIM	Se o campo código não for informado, o <i>WebService</i> vai sugerir o código. Se for informado um tamanho menor do que 6, o <i>WebService</i> completará o código com zeros à esquerda.
Nome	String(254)	Nome do fornecedor	SIM	
Nome Fantasia	String(40)		NÃO	
CNPJCPF	String(14)	CNPJ/CPF do fornecedor	NÃO	
CEP	String(8)	Código Postal do endereço	NÃO	
Logradouro	String(60)	Logradouro do endereço	NÃO	
Bairro	String(30)	Bairro do fornecedor	NÃO	
Numero	String(10)	Número do endereço do fornecedor	NÃO	
Complemento	String(20)	Complemento do endereço	NÃO	
UF	String(2)	Unidade Federativa do endereço	NÃO	
Cidade	String(5)	Código IBGE do município	NÃO	Código IBGE do município sem a parte

				da UF
DDD	String(4)	Código DDD	NÃO	
Fone	String(20)	Número do telefone	NÃO	
Fax	String(20)	Número do fax	NÃO	
Email	String(254)	Endereço de e-mail do fornecedor	NÃO	
Contato	String(30)	Nome do contato	NÃO	
Site	String(60)	Endereço web do fornecedor	NÃO	
RG	String(20)	Número do Registro Geral do Fornecedor	NÃO	Identidade
IE	String(20)	Inscrição Estadual	NÃO	Inscrição Estadual
Indicador IE	Integer	0 - NÃO 1- SIM	NÃO	
Grupo	String(4)	Código do Grupo de Fornecedores	NÃO	Chave da tabela de Grupo de Fornecedores
Receita	String(10)	Código da Despesa	NÃO	Chave da tabela de Receitas/Despesas (Despesa, e não Receita)
IM	String(20)	Inscrição Municipal	NÃO	Inscrição Municipal
Referencia	String(100)	Descrição da Referência	NÃO	Ponto de Referência do Endereço
FinalidadeTed	String(1)	Finalidade da Transferência Eletrônica de Documentos	NÃO	Sem conta – 0 Conta corrente – 1 Poupança – 2
FinalidadeDoc	String(2)	Finalidade do DOC.	NÃO	Valores – 00, 01, 02, 03, 04, 05, 06, 07, 08, 09, 10, 11 ou vazio(""); * Ver Tabela 1
Celular	String(20)	Número do Celular	NÃO	
Leiaute	String(1)	Leiaute FEBRABAN	NÃO	Valores – 1, 2, 3, 4, 5, 6, 7, 8, 9 ou vazio("") * Ver Tabela 2
Banco	String(10)	Código do Banco	NÃO	Código do Banco no AG
BancoNumero	String(10)	Número FEBRABAN do Banco	NÃO	Número FEBRABAN do Banco (Pode ser informado no lugar do código do banco)
Agencia	String(10)	Número da agência	NÃO	Agência
CC	String(10)	Número da conta corrente	NÃO	Conta-Corrente
DVAg	String(1)	Dígito verificador da agência	NÃO	Dígito verificador da Agência
DVCC	String(2)	Dígito verificador da	NÃO	Dígito verificador da

		conta-corrente		Conta-Corrente
ServicoPublico	Integer	Indica se é prestador de serviços público: 0 - NÃO 1 - SIM	NÃO	Se é prestador de serviços públicos
ICMS	Integer	Indica se deve calcular ICMS: 0 - NÃO 1 - SIM	NÃO	Se calcula ICMS
IPI	Integer	Indica se deve calcular IPI: 0 - NÃO 1 - SIM	NÃO	Se calcula IPI
IDWS	String(30)	Indicador do <i>WebService</i>	NÃO	

2.2. Métodos

- **IncluiFornecedor**

- Assinatura:


```
function IncluiFornecedor(const Fornecedor: TFornecedor): String; stdcall;
```
- Parâmetros:
 - Objeto do tipo TFornecedor;
- Possíveis retornos:
 - Código do Fornecedor, caso a inclusão seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **AlteraFornecedor**

- Assinatura:


```
function AlteraFornecedor(const Fornecedor: TFornecedor): String; stdcall;
```
- Parâmetros:
 - Objeto do tipo TFornecedor;
- Possíveis retornos:
 - Código do Fornecedor, caso a alteração seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a alteração;

- **ExcluiFornecedor**

- Assinatura:


```
function ExcluiFornecedor(const Fornecedor: TFornecedor): String; stdcall;
```
- Parâmetros:
 - Objeto do tipo TFornecedor;
- Possíveis retornos:
 - 'OK', caso a exclusão seja bem sucedida;

- Descrição do Erro, caso aconteça algum erro durante a exclusão;

- **GetJsonFornecedorPorCodigo**

- Assinatura:

```
function GetJsonFornecedorPorCodigo(const Codigo: String):
String; stdcall;
```

- Parâmetros:

- Código do Fornecedor;

- Possíveis retornos:

- string no formato JSON, caso o fornecedor seja encontrado;
- string com a mensagem “Fornecedor '000001' não encontrado”;

- **GetFornecedorWithCPFCNPJ**

- Assinatura:

```
function GetFornecedorWithCPFCNPJ(const CNPJCPF: String):
String; stdcall;
```

- Parâmetros:

- CNPJCPF;

- Possíveis retornos:

- Código do fornecedor, caso o fornecedor seja encontrado;
- String vazia (”) , caso o fornecedor não seja encontrado;
- Descrição do Erro, caso aconteça algum erro durante a busca;

- **GetCodigoFornecedorWithIDWS**

- Assinatura:

```
function GetCodigoFornecedorWithIDWS(const IDWS: String):
String; stdcall;
```

- Parâmetros:

- IDWS;

- Possíveis retornos:

- Código do fornecedor, caso o fornecedor seja encontrado;
- Mensagem “ERRO: Fornecedor com IDWS não encontrado.”
- Descrição do Erro, caso aconteça algum erro durante a busca;

- **GetCodigoFornecedorWithIDWSRetornoVazio**

- Assinatura:

```
function GetCodigoFornecedorWithIDWSRetornoVazio(const IDWS:
String): String; stdcall;
```

- Parâmetros:

- IDWS;

- Possíveis retornos:

- Código do fornecedor, caso o fornecedor seja encontrado;
- String vazia (”) , caso o fornecedor não seja encontrado;
- Descrição do Erro, caso aconteça algum erro durante a busca;

- **GetCNPJCPFFornecedorIDWS**

- Assinatura:

```
function GetCNPJCPFFornecedorIDWS(const aIDWS: String): String;
stdcall;
```

- Parâmetros:
 - IDWS;
- Possíveis retornos:
 - CNPJ do fornecedor, caso o fornecedor seja encontrado;
 - Descrição do Erro, caso aconteça algum erro durante a busca;
- **GetListaFornecedores**
 - Assinatura:


```
function GetListaFornecedores: String; stdcall;
```
 - Parâmetros:
 - Não tem parâmetros;
 - Possíveis retornos:
 - String com a lista de fornecedores da empresa;
- **GetListaFornecedoresXML**
 - Assinatura:


```
function GetListaFornecedoresXML: String; stdcall;
```
 - Parâmetros:
 - Não tem parâmetros;
 - Possíveis retornos:
 - string no formato xml com a lista de fornecedores;

3. Serviço

3.1. Classe TService

Campo	Tipo	Descrição	Obrigatório	Observação
Codigo	String(4)	Código do Serviço	SIM	Se o campo código não for informado, o <i>WebService</i> vai sugerir o código. Se for informado um tamanho menor do que 4, o <i>WebService</i> completará o código com zeros à esquerda.
Nome	String(40)	Nome do Serviço	SIM	
NomeApresentar	String(20)	Nome a apresentar do Serviço	NÃO	
Valor	Double	Valor do Serviço	NÃO	
ISS	Integer	Indica se vai reter ISS 0 – NÃO 1 - SIM	SIM	
INSS	Integer	Indica se vai reter INSS 0 – NÃO 1 - SIM	SIM	
IRRF	Integer	Indica se vai reter IRRF 0 – NÃO 1 - SIM	SIM	

PISCOFINSCSL	Integer	Indica se vai reter PIS, COFINS e CSLL 0 – NÃO 1 - SIM	SIM	
AliqISS	Double	Alíquota para cálculo do ISS	NÃO	
AliqINSS	Double	Alíquota para cálculo do INSS	NÃO	
AliqIRRF	Double	Alíquota para cálculo do IRRF	NÃO	
AliqPIS	Double	Alíquota para cálculo do PIS	NÃO	
AliqCOFINS	Double	Alíquota para cálculo do COFINS	NÃO	
AliqCSL	Double	Alíquota para cálculo do CSLL	NÃO	

3.2. Métodos

- **IncluiServico**

- Assinatura:

```
function IncluiServico(const Servico: TServico): String;
stdcall;
```

- Parâmetros:

- Objeto do tipo TServico

- Possíveis retornos:

- Código do Serviço, caso a inclusão seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **AlterarServico**

- Assinatura:

```
function AlterarServico(const Servico: TServico): String;
stdcall;
```

- Parâmetros:

- Objeto do tipo TServico;

- Possíveis retornos:

- Código do Serviço, caso a alteração seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a alteração;

- **ExcluirServico**

- Assinatura:

```
function ExcluirServico(const Servico: TServico): String;
stdcall;
```

- Parâmetros:

- Objeto do tipo TServico;

- Possíveis retornos:

- 'Serviço Excluído', caso a exclusão seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a exclusão;

- **GetListaServicos**

- Assinatura:

```
function GetListaServicos(const EMP_Codigo: string):
TListaServicos; stdcall;
```

- Parâmetros:

- Código da Empresa;
- Possíveis retornos:
 - Lista de serviços da Empresa;
- **GetListaServicosXML**
 - Assinatura:


```
function GetListaServicosXML: String; stdcall;
```
 - Parâmetros:
 - Não tem parâmetros;
 - Possíveis retornos:
 - string no formato xml com a lista de serviços;

4. Item

4.1. Classe TItem

Campo	Tipo	Descrição	Obrigatório	Observação
Codigo	String(10)	Código do item	SIM	Se o campo código não for informado, o <i>WebService</i> vai sugerir o código. Se for informado um tamanho menor do que 10, o <i>WebService</i> completará o código com zeros à esquerda.
Valor	Double	Valor padrão do item	NÃO	Valor Padrão do Item. O valor informado neste campo será utilizado como sugestão nas movimentações deste Item.
Nome	String(60)	Nome do item	SIM	
Unidade	String(3)	Abreviação da unidade de saída	NÃO	
TipolCMSSubst	String(1)	Indica o tipo de substituição do ICMS	SIM	A – Agregação, V – Valor de Pauta
ComplementoDescricao	Blob	Complemento da descrição do item	NÃO	
CodigoutilizadoESTFiscal	String(60)	Mapeamento do item entre o AG e o AC Fiscal	NÃO	Código utilizado na empresa AC Fiscal
CodigoNCM	String(20)	Código NCM do item	NÃO	Nomenclatura Comum do Mercosul
CodigoBarra	String(40)	Código de barras	NÃO	Código Barra GTIN
CodigoBarraUnidadeTribut	String(20)	Código de barras unidade tributável	NÃO	Cód. Barra GTIN Unid. Tributável
CodigoNVE	String(20)	Código NVE do item	NÃO	Nomenclatura de Valor Aduaneiro e Estatística

CodigoANP	String(9)	Código ANP do item	NÃO	Agência Nacional do Petróleo
CodProFortesFiscal	Integer	Mapeamento do item entre o AG e o AC Fiscal	NÃO	Código produto AC Fiscal
RegistroAnvisa	String(20)	Registro do item na anvisa	NÃO	Registro Anvisa
Despesa	String(10)	Código da despesa do item	NÃO	Chave da tabela de Receitas/Despesas Existe uma configuração no sistema para informar a despesa padrão.
Referencia	String(20)	Código de referência do item	NÃO	Referência
UtilizarAliquotaNCM	Integer	0 - NÃO 1 - SIM	SIM	Define se o sistema utilizará as alíquotas padrão ou as alíquotas do cadastro de NCM ao gerar a mensagem da Lei Nº 12.741/2012 ao emitir Notas Fiscais.
ClassificacaoFiscal	String(10)	Código de classificação fiscal do item	NÃO	Classificação Fiscal
DescricaoUnidade	String(60)	Descrição da unidade	NÃO	Descrição da Unidade de Medida.
CalculaICMS	Integer	Indica se o sistema deve calcular ICMS para este item: 0 - NÃO 1 - SIM	SIM	
DeducaoICMS	Integer	Indica se deve haver dedução de ICMS para este item: 0 - NÃO 1 - SIM	SIM	
CalculaIRRF	Integer	Indica se o sistema deve calcular Imposto de Renda: 0 - NÃO 1 - SIM	SIM	
CalculaPIS	Integer	Indica se o sistema deve calcular PIS: 0 - NÃO 1 - SIM	SIM	
CalculaCOFINS	Integer	Indica se o sistema deve calcular COFINS: 0 - NÃO 1 - SIM	SIM	
CalculaCSL	Integer	Indica se o sistema deve calcular CSLL: 0 - NÃO 1 - SIM	SIM	
AliquotaIPI	Double	Percentual da alíquota de	NÃO	

		IPI		
SitTribIPISaida	String(2)	Código da situação tributária do IPI	NÃO	Situação tributária do IPI nas Saídas
SitTribCOFINSSaida	String(2)	Código da situação tributária do COFINS	NÃO	Situação Tributária do COFINS nas Saídas
SitTribPISsaida	String(2)	Código da situação tributária do PIS	NÃO	Situação Tributária do PIS nas Saídas
NaturezaReceitaCOFINS	Integer	Identificador da tabela de Natureza da Receita COFINS	NÃO	Sequencial da Natureza da Receita COFINS
NaturezaReceitaPIS	Integer	Identificador da tabela de Natureza da Receita PIS	NÃO	Sequencial da Natureza da Receita PIS
ICMSSubstituicao	Integer	Indica se o ICMS é ou não de substituição: 0 - NÃO 1 - SIM	SIM	ICMS Substituição
AliquotalCMSSubstAgregacao	Double	Percentual da alíquota de ICMS de substituição por agregação	NÃO	Alíquota de ICMS de Substituição por Agregação
ValorPautaICMSSubst	Double	Valor de pauta do ICMS de substituição	NÃO	Valor de Pauta do ICMS de Substituição
IPIFixo	Integer	Indica se o item possui IPI fixo: 0 - NÃO 1 - SIM	SIM	Se o IPI do Item é fixo
ValorIPIFixo	Double	Valor fixo do IPI	NÃO	Valor Fixo do IPI
Categoria	String(4)	Código da categoria	NÃO	Chave da tabela de Categorias
Acabamento	String(4)	Código do Acabamento	NÃO	Chave da tabela de Acabamentos
Material	String(4)	Código do Material	NÃO	Chave da tabela de Materiais
UnidadeEntrada	String(3)	Unidade de Entrada do Item	NÃO	
Grupo	String(4)	Código de Grupo do Item	NÃO	
Tipo	String(4)	Código de Tipo do Item	NÃO	
IPI	Integer	Indica se deve calcular IPI: 0 - NÃO 1 - SIM	SIM	Se calcula IPI
STBSNCodigoSaida	String(3)		NÃO	Campo usado para sugerir a Situação Tributária B nas movimentações
STBSNCodigoForaEstadoSaida	String(3)		NÃO	
STBSNCodigoForaPaísSaida	String(3)		NÃO	
STBSNCodigoNFPropria	String(3)		NÃO	
STBSNForaEstadoNFPropria	String(3)		NÃO	

STBSNForaPaisNFPropria	String(3)		NÃO	
STBSNCodigoNFEEntada	String(3)		NÃO	
STBSNCodigoForaEstadoNFEEntada	String(3)		NÃO	
STBSNCodigoForaPaisNFEEntada	String(3)		NÃO	
IDWS	String(30)	Indicador do Web Service	NÃO	

4.2. Métodos

- **IncluiItem**

- Assinatura:
`function IncluiItem(const Item: TItem): String; stdcall;`
- Parâmetros:
 - Objeto do tipo TItem
- Possíveis retornos:
 - Código do Item, caso a inclusão seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **IncluiArrayItem**

- Assinatura:
`function IncluiArrayItem(ArrayItem: TArrayItem): TArrayItem; stdcall;`
- Parâmetros:
 - Objeto do tipo TArrayItem
- Possíveis retornos:
 - Array de itens, caso a inclusão seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **AlterarItem**

- Assinatura:
`function AlterarItem(const Item: TItem): String; stdcall;`
- Parâmetros:
 - Objeto do tipo TItem
- Possíveis retornos:
 - Código do Item, caso a alteração seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a alteração;

- **AlterarArrayItem**

- Assinatura:
`function AlterarArrayItem(ArrayItem: TArrayItem): TArrayItem; stdcall;`
- Parâmetros:
 - Objeto do tipo TArrayItem;
- Possíveis retornos:
 - Array de itens, caso a alteração seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a alteração;

- **ExcluiItem**
 - Assinatura:


```
function ExcluiItem(const Item: TItem): String; stdcall;
```
 - Parâmetros:
 - Objeto do tipo TItem
 - Possíveis retornos:
 - 'OK', caso a exclusão seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a exclusão;
- **ExcluiArrayItem**
 - Assinatura:


```
function ExcluiArrayItem(ArrayItem: TArrayItem): String; stdcall;
```
 - Parâmetros:
 - Objeto do tipo TArrayItem;
 - Possíveis retornos:
 - Array de itens, caso a exclusão seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a exclusão;
- **GetJsonItemPorCodigo**
 - Assinatura:


```
function GetJsonItemPorCodigo(const Codigo: String): String; stdcall;
```
 - Parâmetros:
 - Código do Item;
 - Possíveis retornos:
 - String no formato JSON com as informações do item passado por parâmetro;
- **GetObjetoItemPorCodigo**
 - Assinatura:


```
function GetObjetoItemPorCodigo(const Codigo: String): TItem; stdcall;
```
 - Parâmetros:
 - Código do Item;
 - Possíveis retornos:
 - Objeto do tipo TItem, caso exista item com o código passado;
- **GetListaltensPorGrupo**
 - Assinatura:


```
function GetListaItensPorGrupo(GIT_Codigo: String): TlistaItens; stdcall;
```
 - Parâmetros:
 - Código do Grupo do Item;
 - Possíveis retornos:
 - Objeto do tipo TListaltens, caso exista itens com o código do grupo de item passado;
- **GetListaltensPorCentroEstoque**

- Assinatura:


```
function GetListaItensPorCentroEstoque(CES_Codigo: String): TlistaItens;stdcall;
```
 - Parâmetros:
 - Código do Centro de Estoque;
 - Possíveis retornos:
 - Objeto do tipo TListaltens, caso exista itens com o código do centro de estoque passado;
- **GetListaltensPorCentroEstoque**
 - Assinatura:


```
function GetListaItensPorCentroEstoqueXML(CES_Codigo: String): String;stdcall;
```
 - Parâmetros:
 - Código do Centro de Estoque;
 - Possíveis retornos:
 - String no formato XML, caso exista itens com o código do centro de estoque passado;
- **GetListaltensCustoMedio**
 - Assinatura:


```
function GetListaCustoMedio(PRO_Codigo: String = ''; GIT_Codigo: String = ''; CES_Codigo: String = ''):TlistaItens;stdcall;
```
 - Parâmetros(Opcionais):
 - PRO_Codigo(Código do Item), GIT_Codigo(Código do grupo de Itens) e CES_Codigo (Código do centro de estoque);
 - Possíveis retornos:
 - Objeto do tipo TListaltens, caso exista itens com os parâmetros passados;
- **GetListaltensCustoMedioJSON**
 - Assinatura:


```
function GetListaCustoMedioJSON(PRO_Codigo: String = ''; GIT_Codigo: String = ''; CES_Codigo: String = ''):String;stdcall;
```
 - Parâmetros(Opcionais):
 - PRO_Codigo(Código do Item), GIT_Codigo(Código do grupo de Itens) e CES_Codigo (Código do centro de estoque);
 - Possíveis retornos:
 - String no formato JSON com as informações do item passado por parâmetro.
- **GetListaGruposdeltensArray**
 - Assinatura:


```
function GetListaGruposdeItensArray:TlistaGruposdeItens;stdcall;
```
 - Possíveis retornos:
 - Objeto do tipo TlistaGruposdeltens com listagem de grupos de itens.
- **GetListaGruposdeltensJSON**
 - Assinatura:


```
function GetListaGruposdeItensJSON:String;stdcall;
```
 - Possíveis retornos:
 - String no formato JSON com listagem de grupos de itens.

5. Lançamentos

5.1. Classe TLancamentos

Campo	Tipo	Descrição	Obrigatório	Observação
Codigo	String(11)	Código do Lançamento	SIM	Gerado pelo <i>WebService</i>
Estabelecimento	String(4)	Código do Estabelecimento	SIM	
Conta	String(15)	Código da Conta Contábil	SIM	
Data	XSDatetime	Data do Lançamento	SIM	Data do Lançamento
Sequencial	Integer	Sequencial do Lançamento na data	SIM	Gerado pelo <i>WebService</i>
CentroResultados	String(10)	Código do Centro de Resultados	Obrigatório se não houver rateios	
ReceitaDespesa	String(10)	Código da Receita/Despesa	Obrigatório se não houver rateios	
Valor	Double	Valor do Lançamento	SIM	
Natureza	String(1)	Natureza do Lançamento: E - Entrada S - Saída	SIM	
ExportaAC	Integer	Indica se deve exportar o Lançamento para o AC Contábil: 0 - NÃO 1 - SIM	SIM	Caso não preencha o sistema coloca '0-Não'
Historico	Blob	Histórico do Lançamento	NÃO	
RateioLancamento	TArrayRateioLancamentos	Rateios do Lançamento	NÃO	Array de objetos do tipo <i>TRateioLancamentos</i>

5.2. Classe TRateioLancamentos

Campo	Tipo	Descrição	Obrigatório	Observação
Codigo	String(11)	Código do Lançamento	SIM	Chave estrangeira para o Lançamento
Sequencial	Integer	Identificador do Rateio	SIM	
Estabelecimento	String(4)	Código do Estabelecimento	SIM	
CentroResultados	String(10)	Código do Centro de Resultados	SIM	
ReceitaDespesa	String(15)	Código da Receita/Despesa	SIM	

Natureza	String(1)	Natureza do Rateio: E - Entrada S - Saída	NÃO	
Valor	Double	Valor do Rateio	SIM	
Historico	Blob	Histórico do Rateio		

5.3. Métodos

- **IncluiLancamento**

- Assinatura:

```
function IncluiLancamento(const Lancamento: TLancamentos): string; stdcall;
```

- Parâmetros:

- Objeto do tipo TLancamentos;

- Possíveis retornos:

- Código do Lançamento, caso a inclusão seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **ExcluiLancamento**

- Assinatura:

```
function ExcluiLancamento(const Codigo: string): string; stdcall;
```

- Parâmetros:

- Código do Lançamento;

- Possíveis retornos:

- 'OK', caso a exclusão seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a exclusão;

- **GetObjetoLancamento**

- Assinatura:

```
function GetObjetoLancamento(const CodLancamento: String): TLancamentos; stdcall;
```

- Parâmetros:

- Código do Lançamento;

- Possíveis retornos:

- Objeto do tipo TLancamentos, caso a busca seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a busca;

- **GetObjetoLancamentoIDWS**

- Assinatura:

```
function GetObjetoLancamentoIDWS(const IDWS: String): TLancamentos; stdcall;
```

- Parâmetros:

- IDWS;

- Possíveis retornos:

- Objeto do tipo TLancamentos, caso a busca seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a busca;

6. Contas a Receber

6.1. TContasaReceber

Código	Tipo	Descrição	Obrigatório	Observação
codigo	String(11)	Código do Contas a Receber	SIM	Utilizado somente para consulta. Se for informado na inclusão, o valor deste campo será ignorado.
estabelecimento	String(4)	Código do estabelecimento	SIM	
centroResultados	String(10)	Código do Centro de Resultados	-	Obrigatório se não for informar rateio
cliente	String(14)	CPF/CNPJ do Cliente	SIM	
receita	String(10)	Código da Receita	-	Obrigatório se não for informar rateio
documento	String(20)	Número do Documento	NÃO	
tipoGeracao	String(1)	Tipo de Geração do Contas a Receber: A - Automático M - Manual	SIM	Deve ser informado como M - Manual.
emissao	XSDatetime	Data da Emissão do Contas a Receber	SIM	
exportaAC	Integer	Indica se o Contas a Receber deve ser exportado para o sistema AC. 0 - Não 1 - Sim	NÃO	Ao gerar o arquivo de exportação para o Fortes Contábil, o sistema verifica se o registro deve ou não ser exportado.
ISS	Double	Valor do ISS	NÃO	
IRRF	Double	Valor do IRRF	NÃO	
INSS	Double	Valor do INSS	NÃO	
PIS	Double	Valor do PIS	NÃO	
DescontoIncondicional	Double	Valor do Desconto Incondicional	NÃO	
COFINS	Double	Valor do COFINS	NÃO	
CSLL	Double	Valor do CSLL	NÃO	
Obs	Blob	Texto contendo uma observação.	NÃO	
mesAno	String(6)	Mês/Ano de Competência do Contas a Receber	NÃO	
vencimentosaReceber	Array de Objetos	Lista de vencimentos diversos	(*)	TArrayOfVencimento saReceber
vencimentosChequeaReceber	Array de Objetos	Lista de vencimentos de cheque	(*)	TArrayOfVencimento sChequeaReceber
vencimentosCartaoaReceber	Array de Objetos	Lista de vencimentos de cartão	(*)	TArrayOfVencimento sCartaoaReceber
servicosaReceber	Array de Objetos	Lista de serviços do Contas a Receber	(*)	TArrayOfServicosaReceber
itensaReceber	Array de Objetos	Lista de Itens do Contas a Receber	(*)	TArrayOfItensaReceber
rateiosaReceber	Array de Objetos	Lista de Rateios do Contas a Receber	(**)	TArrayOfRateiosaReceber
IDWS	String(30)		NÃO	

ClienteIDWS	String(30)		NÃO	
CampoExtra1	String(254)		NÃO	
CampoExtra2	String(254)		NÃO	
CampoExtra3	String(254)		NÃO	
CampoExtra4	String(254)		NÃO	
CampoExtra5	String(254)		NÃO	
CampoExtra6	String(254)		NÃO	
CampoExtra7	String(254)		NÃO	
CampoExtra8	String(254)		NÃO	
CampoExtra9	String(254)		NÃO	
CampoExtra10	String(254)		NÃO	
ContaAplicacao	String(30)		NÃO	Chave estrangeira para Contas de Aplicacao
GRECODIGO	String(4)		NÃO	Chave estrangeira para Grupo de Contas a Receber
OutrasRetencoes	Double		NÃO	
HistoricoPadrao	String(5)		NÃO	Chave estrangeira para Histórico Padrão
ContaContabilCliente	String(15)		NÃO	

(*) (obrigatório ter ServicosaReceber OU ItensaReceber) E (VencimentosDiversos OU VencimentosCartao OU VencimentosCheque)

(**) Obrigatório caso “Receita” e/ou “Centro de Resultados” não forem informados

6.2. TVencimentosaReceber

Código	Tipo	Descrição	Obrigatório	Observação
codigo	String(11)	Código do Contas a Receber	SIM	Somente para informação. Na inclusão será utilizado o Código do Objeto Pai (TContasaReceber)
sequencial	Integer	Sequencial de identificação do vencimento	SIM	
vencimento	XSDatetime	Data do vencimento	SIM	
valor	Double	Valor do vencimento	SIM	
ValorOriginal	Double		SIM	
tipoDocumento	String(3)	Código do Tipo de Documento	NÃO	
titulo	String(20)	Título do vencimento	NÃO	
agenteCobrador	String(4)	Código do agente cobrador	SIM	
nrBoleto	String(20)	Número do boleto	NÃO	
idBloquete	Integer	Identificação do Bloquete	NÃO	
descontoPrevisto	Double	Valor do Desconto Previsto	NÃO	
dtCancelamento	XSDatetime	Data do cancelamento do Vencimento	NÃO	

Quitado	Integer	Indica se o vencimento está quitado 0 - Não 1 - SIM	NÃO	
Obs	Blob		NÃO	
FormaPagamento	String(4)		NÃO	Chave estrangeira para Forma de Pagamento
CodigoBarras	String(44)		NÃO	
LinhaDigitada	String(100)		NÃO	
PercentualJuros	Double		NÃO	
PercentualMulta	Double		NÃO	

6.3. TVencimentosChequeaReceber

Código	Tipo	Descrição	Obrigatório	Observação
codigo	String(11)	Código do Contas a Receber	SIM	Somente para informação. Na inclusão será utilizado o Código do Objeto Pai (TContasaReceber)
sequencial	Integer	Sequencial identificador do vencimento	SIM	
vencimento	XSDatetime	Data do vencimento	SIM	
valor	Double	Valor do vencimento	SIM	
codBanco	String(4)	Código do banco no AG	NÃO	Chave para a tabela de bancos. Não é obrigatório se informar o campo bancoNumero
agencia	String(10)	Número da Agência	NÃO	
cc	String(10)	Número da Conta Corrente	NÃO	
numero	String(10)	Número do Cheque	NÃO	
titular	String(40)	Nome do Titular do Cheque	NÃO	
bancoNumero	String	Número do Banco FEBRABAN	NÃO	Campo utilizado para encontrar o código do banco no sistema.
quitado	Integer	Indica se o vencimento está quitado 0 - Não 1 - SIM	NÃO	
Titulo	String(20)		NÃO	
Obs	Blob		NÃO	

6.4. TVencimentosCartaoaReceber

Código	Tipo	Descrição	Obrigatório	Observação
codigo	String(11)	Código do Contas a	SIM	Somente para

		Receber		informação. Na inclusão será utilizado o Código do Objeto Pai(TContasaReceber)
sequencial	Integer	Identificador do Vencimento	SIM	
administradora	String(4)	Código da administradora do cartão	SIM	Chave para a tabela de administradoras De cartões
vencimento	XSDatetime	Data do vencimento	SIM	
valor	Double	Valor do vencimento	SIM	
TAD_Valor	Double	Valor da Taxa da administradora de cartões		
quitado	Integer	Indica se o vencimento está quitado 0 - Não 1 - Sim	SIM	
Bandeira	String(20)	Código da bandeira	NÃO	Este campo será utilizado junto com o campo Rede para consultar a Administradora cadastrada no sistema.
Rede	String(20)	Código da rede	NÃO	Este campo será utilizado junto com o campo Rede para consultar a Administradora cadastrada no sistema.
ReferenciaADM	String(30)	Campo de Referência da Administradora.	NÃO	Este campo será utilizado para pesquisa da Administradora, caso o campo Código ou os campos Rede e Bandeira não sejam informados.
NSU	String(30)		NÃO	
NRAutorizacao	String(15)		NÃO	
Titulo	String(20)		NÃO	
Obs	Blob		NÃO	
IDTransaction	String(35)		NÃO	

6.5. TitensaReceber

Código	Tipo	Descrição	Obrigatório	Observação
Codigo	String(11)	Código do Contas a Receber	SIM	Somente para informação. Na

				inclusão será utilizado o Código do Objeto Pai (TContasaReceber)
Produto	String(10)	Código do Item	SIM	
Sequencial	Integer	Sequencial de identificação do item a receber	SIM	
Quantidade	Double	Quantidade dos itens	SIM	
Valor	Double	Valor unitário do item	SIM	
CFOP_Codigo	String(4)	Código do CFOP do Item	NÃO	Chave da tabela de CFOP (é o próprio CFOP)
STA_Codigo	String(1)	Código da situação tributária A	NÃO	Chave da tabela de Situação Tributária A
STB_Codigo	String(2)	Código da situação tributária B	NÃO	Chave da tabela de Situação Tributária B
AliquotaICMS	Double	Percentual da alíquota de ICMS	NÃO	
AliquotaIPI	Double	Percentual da alíquota do IPI	NÃO	
Unidade	String(3)	Unidade de medida	NÃO	
CompDescricao	Blob	Complemento da descrição do item	NÃO	
AliqICMSSubst	Double	Percentual da alíquota do ICMS de substituição	NÃO	
BaseICMSSubst	Double	Base de cálculo do ICMS de substituição	NÃO	
ValorICMSSubst	Double	Valor do ICMS de substituição	NÃO	
Frete	Double	Valor do Frete	NÃO	
ComissionadosItensReceber	TArrayOfComissionadosItensReceber		NÃO	
Desconto	Double		NÃO	

6.5.1. TComissionadosItensReceber

Código	Tipo	Descrição	Obrigatório	Observação
Codigo_CRE	String(11)	Código do Contas a Receber	SIM	Somente para informação. Na inclusão será utilizado o Código do Objeto Pai (TContasaReceber)
ICR_Sequencial	Integer	Sequencial do item a receber	SIM	
VendedorRepresentante	String(4)	Código do vendedor/representante	SIM	
Percentual	Double	Percentual de comissão	SIM	
Sequencial	Integer	Identificador do comissionado	SIM	

6.6. TServiceaReceber

Código	Tipo	Descrição	Obrigatório	Observação
Codigo	String(11)	Código do Contas a Receber	SIM	Somente para informação. Na inclusão será utilizado o Código do Objeto Pai (TContasaReceber)
Sequencial	Integer	Identificador do serviço a receber	SIM	
Servico	String(4)	Código do serviço	SIM	
Modalidade	String(4)	Código da modalidade	SIM	
Valor	Double	Valor Líquido do serviço	SIM	
DescontoIncondicional	Double	Valor do desconto incondicional	NÃO	
ValorBruto	Valor	Valor bruto do serviço	SIM	
ComissionadosServicosaReceber	Array de Objetos	Lista com os dados dos vendedores/representantes comissionados nos serviços	NÃO	TComissionadosServicosaReceber
AliquotaISS	Double		NÃO	
Obs	Blob		NÃO	
Quantidade	Double	Quantidade de serviços	NÃO	Se a configuração para utilizar a quantidade de serviços não estiver marcada, a quantidade informada será ignorada. - O valor bruto será considerado como valor unitário do serviço. - Será considerado como 1, caso não seja informado.

6.6.1. TComissionadosServicosaReceber

Código	Tipo	Descrição	Obrigatório	Observação
Codigo_CRE	String(11)	Código do Contas a Receber	SIM	Somente para informação. Na inclusão será utilizado o Código do Objeto Pai (TContasaReceber)
SCR_Sequencial	Integer	Sequencial do serviço a receber	SIM	
VendedorRepresentante	String(4)	Código do vendedor/representante	SIM	
Percentual	Double	Percentual de comissão	NÃO	
Sequencial	Integer	Identificador do comissionado	NÃO	

6.7. TBaixaContasaReceber

Código	Tipo	Descrição	Obrigatório	Observação
CRE_Codigo	String(11)	Código do Contas a Receber	SIM	
SequencialVencimento	Integer	Sequencial de identificação do Vencimento	SIM	
Sequencial	Integer	Sequencial da Baixa	SIM	
ExportaAC	Integer	Indicação se exporta a baixa para o AC Contábil		0 -Não 1 - Sim
Data	XSDatetime	Data da Baixa	SIM	
Valor	Double	Valor da Baixa	SIM	
Desconto	Double	Valor do Desconto	NÃO	
Juros	Double	Valor dos Juros	NÃO	
Multa	Double	Valor da Multa	NÃO	
LAN_ContaFinanceira	String(4)	Chave da Conta Financeira	SIM	
Lan_Historico	Text	Histórico do Lançamento	NÃO	
LAN_CRD_Desconto	String(10)	Código da Despesa do Desconto	NÃO	
LAN_CRD_Tarifa	String(10)		NÃO	
LAN_CRD_Juros	String(10)	Código da Receita de Juros	NÃO	
LAN_CRD_Multa	String(10)	Código da Receita de Multa	NÃO	

6.8. TBaixaVencimentoCartaoaReceber

Código	Tipo	Descrição	Obrigatório	Observação
CRE_Codigo	String(11)	Código do Contas a Receber	SIM	
SequencialVencimento	Integer	Sequencial de identificação do Vencimento	SIM	
Sequencial	Integer	Sequencial da Baixa	SIM	
Data	XSDatetime	Data da Baixa	SIM	
Valor	Double	Valor da Baixa	SIM	
Taxa	Double	Valor da Taxa	NÃO	
DtCancel	XSDatetime	Data do Cancelamento	NÃO	Somente para consulta
MotivoCancel	Blob	Motivo do Cancelamento	NÃO	Somente para consulta
LAN_ContaFinanceira	String(4)	Chave da Conta Financeira	NÃO	
LAN_Historico	Texto	Histórico do Lançamento	NÃO	
LAN_CRD_Taxa	String(10)	Código da Despesa da Taxa Adm. de Cartões	NÃO	

6.9. TRateiosaReceber

Código	Tipo	Descrição	Obrigatório	Observação
CRE_Codigo	String(11)	Código do Contas a Receber	SIM	
Estabelecimento	String(4)		SIM	
CentroDeResultados	String(10)		SIM	
Receita	String(10)		SIM	
Observacao	Blob		NÃO	
Valor	Double		NÃO	Se não for informado será incluído ZERO

6.10. Métodos

- **IncluiContasaReceber**

- Assinatura:

- ```
function IncluiContasaReceber(const ContasaReceber: TContasaReceber): String; stdcall;
```

- Parâmetros:

- Objeto do tipo TContasaReceber;

- Possíveis retornos:

- Código do Contas a Receber, caso a inclusão tenha sido bem sucedida;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **AlteraContasaReceber**

- Assinatura:

- ```
function AlteraContasaReceber(const ContasaReceber: TContasaReceber; const Codigo: String): String; stdcall;
```

- Parâmetros:

- Código do Contas a Receber;
 - Objeto do tipo TContasaReceber;

- Possíveis retornos:

- Código do Contas a Receber, caso a alteração tenha sido bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a alteração;

- **ExcluiContasaReceber**

- Assinatura:

- ```
function ExcluiContasaReceber(const Codigo: String): String; stdcall;
```

- Parâmetros:

- Código do Contas a Receber;

- Possíveis retornos:

- 'OK', caso a exclusão tenha sido bem sucedida;
    - 'Contas a Receber não cadastrado', caso o contas a receber passado não seja encontrado;
    - Descrição do Erro, caso aconteça algum erro durante a exclusão;

- **ExcluirContasaReceberComIDWS**

- Assinatura:
 

```
function ExcluirContasaReceberComIDWS(const IDWS: String):
String; stdcall;
```
  - Parâmetros:
    - IDWS;
  - Possíveis retornos:
    - 'OK', caso a exclusão tenha sido bem sucedida;
    - 'Contas a Receber não cadastrado', caso o contas a receber passado não seja encontrado;
    - Descrição do Erro, caso aconteça algum erro durante a exclusão;
- **GetCodigoContasaReceberWithIDWS**
  - Assinatura:
 

```
function GetCodigoContasaReceberWithIDWS(const IDWS: String):
String; stdcall;
```
  - Parâmetros:
    - IDWS;
  - Possíveis retornos:
    - Código do Contas a Receber, caso exista contas a receber com o IDWS passado;
    - Descrição do Erro, caso aconteça algum erro durante a busca;
- **IncluiBaixaContasReceber**
  - Assinatura:
 

```
function IncluiBaixaContasReceber(const BaixaContasaReceber:
TBaixaContasaReceber): String; stdcall;
```
  - Parâmetros:
    - Objeto do tipo TBaixaContasaReceber;
  - Possíveis retornos:
    - 'OK', caso a inclusão tenha sido bem sucedida;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão;
- **IncluiBaixaContasReceberCheque**
  - Assinatura:
 

```
function IncluiBaixaContasReceberCheque(const
BaixaVencimentoChequeaReceber: TBaixaVencimentoChequeaReceber):
String; stdcall;
```
  - Parâmetros:
    - Objeto do tipo TBaixaVencimentoChequeaReceber;
  - Possíveis retornos:
    - 'OK', caso a inclusão tenha sido bem sucedida;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão;
- **IncluiBaixaContasReceberCartao**
  - Assinatura:
 

```
function IncluiBaixaContasReceberCartao(const
BaixaVencimentoCartaoaReceber: TBaixaVencimentoCartaoaReceber):
String; stdcall;
```
  - Parâmetros:
    - Objeto do tipo TBaixaVencimentoCartaoaReceber;



- Possíveis retornos:
    - 'OK', caso a inclusão tenha sido bem sucedida;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão;
- **CancelaBaixaContasReceber**
  - Assinatura:
 

```
function CancelaBaixaContasReceber(CodigoCRE: String;
SequencialVencimento, Sequencial: Integer; DataCancelamento:
TDateTime; MotivoCancelamento: String): String; stdcall;
```
  - Parâmetros:
    - Código do contas a receber;
    - Sequencial do vencimento;
    - Sequencial;
    - Data do Cancelamento;
    - Motivo do Cancelamento;
  - Possíveis retornos:
    - 'OK', caso o cancelamento tenha sido bem sucedido;
    - 'ERRO: Baixa do Vencimento do Contas a Receber não cadastrada', caso a baixa não tenha sido encontrada;
    - Descrição do Erro, caso aconteça algum erro durante o cancelamento;
- **CancelaBaixaContasReceberIDWS**
  - Assinatura:
 

```
function CancelaBaixaContasReceberIDWS(CancelaBaixaVencimento:
TCancelaBaixaVencimento): String; stdcall;
```
  - Parâmetros:
    - Objeto do tipo TCancelaBaixaVencimento;
  - Possíveis retornos:
    - 'OK', caso o cancelamento tenha sido bem sucedido;
    - 'ERRO:Baixa do Vencimento do Contas a Receber não cadastrada', caso a baixa não tenha sido encontrada;
    - Descrição do Erro, caso aconteça algum erro durante o cancelamento;
- **CancelaBaixaContasReceberCartao**
  - Assinatura:
 

```
function CancelaBaixaContasReceberCartao(CodigoCRE: String;
SequencialVencimento, Sequencial: Integer; DataCancelamento:
TDateTime; MotivoCancelamento: String): String; stdcall;
```
  - Parâmetros:
    - Código do contas a receber;
    - Sequencial do vencimento;
    - Sequencial;
    - Data do Cancelamento;
    - Motivo do Cancelamento;
  - Possíveis retornos:
    - 'OK', caso o cancelamento tenha sido bem sucedido;
    - 'ERRO:Baixa do Vencimento de Cartão do Contas a Receber não cadastrada', caso a baixa não tenha sido encontrada;
    - Descrição do Erro, caso aconteça algum erro durante o cancelamento;

- **CancelaBaixaContasReceberCartaoIDWS**

- Assinatura:

```
function CancelaBaixaContasReceberCartaoIDWS
(CancelaBaixaVencimento: TCancelaBaixaVencimento): String;
stdcall;
```

- Parâmetros:

- Objeto do tipo TCancelaBaixaVencimento;

- Possíveis retornos:

- 'OK', caso o cancelamento tenha sido bem sucedido;
    - 'Baixa do Vencimento de Cartão do Contas a Receber não cadastrada', caso a baixa não tenha sido encontrada;
    - Descrição do Erro, caso aconteça algum erro durante o cancelamento;

- **CancelaBaixaContasReceberCheque**

- Assinatura:

```
function CancelaBaixaContasReceberCheque(CodigoCRE: String;
SequencialVencimento, Sequencial: Integer; DataCancelamento:
TDateTime; MotivoCancelamento: String): String; stdcall;
```

- Parâmetros:

- Código do contas a receber;
    - Sequencial do vencimento;
    - Sequencial;
    - Data do Cancelamento;
    - Motivo do Cancelamento;

- Possíveis retornos:

- 'OK', caso o cancelamento tenha sido bem sucedido;
    - 'Baixa do Vencimento de Cheque do Contas a Receber não cadastrada', caso a baixa não tenha sido encontrada;
    - Descrição do Erro, caso aconteça algum erro durante o cancelamento;

- **CancelaBaixaContasReceberChequeIDWS**

- Assinatura:

```
function CancelaBaixaContasReceberChequeIDWS
(CancelaBaixaVencimento: TCancelaBaixaVencimento): String;
stdcall;
```

- Parâmetros:

- Objeto do tipo TCancelaBaixaVencimento;

- Possíveis retornos:

- 'OK', caso o cancelamento tenha sido bem sucedido;
    - 'Baixa do Vencimento de Cartão do Contas a Receber não cadastrada', caso a baixa não tenha sido encontrada;
    - Descrição do Erro, caso aconteça algum erro durante o cancelamento;

- **CancelaVencimentosDiversosEmAberto**

- Assinatura:

```
function CancelaVencimentosDiversosEmAberto(const
CodContasReceber: String; sDtCancel: TDateTime; aMotCancel:
String):String; stdcall;
```

- Parâmetros:

- Código do contas a receber;
    - Data do cancelamento;
    - Motivo cancelamento;
  - Possíveis retornos:
    - String vazia (") ou a mensagem 'Boletos que possuem baixas (Não foram cancelados): ', caso o cancelamento tenha sido bem sucedido;
    - 'Contas a Receber não cadastrado.', caso o vencimento não tenha sido encontrado;
    - Descrição do Erro, caso aconteça algum erro durante o cancelamento;
- 
- **CancelaVencimentosCartaoaReceberEmAberto**
    - Assinatura:
 

```
function CancelaVencimentosCartaoaReceberEmAberto(const
CodContasReceber: String; sDtCancel: TDateTime; aMotCancel:
String):String; stdcall;
```
    - Parâmetros:
      - Código do contas a receber;
      - Data do cancelamento;
      - Motivo cancelamento;
    - Possíveis retornos:
      - String vazia (") ou a mensagem 'Cartões que possuem baixas (Não foram cancelados): ', caso o cancelamento tenha sido bem sucedido;
      - 'Contas a Receber não cadastrado.', caso o vencimento não tenha sido encontrado;
      - Descrição do Erro, caso aconteça algum erro durante o cancelamento;
- 
- **CancelaVencimentosChequeaReceberEmAberto**
    - Assinatura:
 

```
function CancelaVencimentosChequeaReceberEmAberto(const
CodContasReceber: String; sDtCancel: TDateTime; aMotCancel:
String):String; stdcall;
```
    - Parâmetros:
      - Código do contas a receber;
      - Data do cancelamento;
      - Motivo cancelamento;
    - Possíveis retornos:
      - String vazia (") ou a mensagem 'Cheques que possuem baixas (Não foram cancelados): ', caso o cancelamento tenha sido bem sucedido;
      - 'Contas a Receber não cadastrado.', caso o vencimento não tenha sido encontrado;
      - Descrição do Erro, caso aconteça algum erro durante o cancelamento;
- 
- **AlteraVencimentoDiversoEmAberto**
    - Assinatura:
 

```
function AlteraVencimentoDiversoEmAberto(CodContasReceber,
aOldNrBoleto, aNewNrBoleto: String; aOldIdBloquete,
aNewIdBloquete: Integer; aNewDtVenc: TDateTime): String; stdcall;
```
    - Parâmetros:
      - Código do contas a receber;
      - Antigo número do boleto;
      - Novo número do boleto;

- Antigo número do bloquete;
  - Novo número do bloquete;
  - Nova data do vencimento;
- Possíveis retornos:
  - String vazia (") caso a alteração tenha sido bem sucedida;
  - 'Boleto inexistente.', caso o boleto não tenha sido encontrado;
  - 'Boletos que possuem baixas: ' caso existam boletos com baixa que não serão alterados;
  - 'ERRO: Contas a Receber não cadastrado.', caso o vencimento não tenha sido encontrado;
  - Descrição do Erro, caso aconteça algum erro durante a alteração;
- **AlteraDataVencimentoDiversoEmAberto**
  - Assinatura:
 

```
function AlteraDataVencimentoDiversoEmAberto(CodigoCRE,
NovaDataVenc: String; SequencialVenc: Integer): String; stdcall;
```
  - Parâmetros:
    - Código do contas a receber;
    - Nova data do vencimento;
    - Sequencial do vencimento;
  - Possíveis retornos:
    - Sequencial do Vencimento, caso a alteração tenha sido bem sucedida;
    - 'Contas a Receber não cadastrado.', caso o contas a receber não tenha sido encontrado;
    - Descrição do Erro, caso aconteça algum erro durante a alteração;
- **AlteraDataVencimentoDiversoEmAbertoComIDWSCRE**
  - Assinatura:
 

```
function AlteraDataVencimentoDiversoEmAbertoComIDWSCRE(IDWSCRE,
NovaDataVenc: String; SequencialVenc: Integer): String; stdcall;
```
  - Parâmetros:
    - IDWS do Contas a Receber;
    - Nova data do vencimento;
    - Sequencial do vencimento;
  - Possíveis retornos:
    - Sequencial do Vencimento, caso a alteração tenha sido bem sucedida;
    - 'Contas a Receber não cadastrado.', caso o contas a receber não tenha sido encontrado;
    - Descrição do Erro, caso aconteça algum erro durante a alteração;
- **EnviaCRECodigoContabilidade**
  - Assinatura:
 

```
function EnviaCRECodigoContabilidade(CodigoCRE: String): String;
stdcall;
```
  - Parâmetros:
    - Código do Contas a Receber;
  - Possíveis retornos:
    - Código do Contas a Receber, caso o envio tenha sido bem sucedido;
    - 'ERRO:Contas a Receber não encontrado.', caso o contas a receber não tenha sido encontrado;
    - Descrição do Erro, caso aconteça algum erro durante o envio para contabilidade;

- **EnviaCREIDWSContabilidade**

- Assinatura:

```
function EnviaCREIDWSContabilidade(CodigoIDWSCRE: String):
String; stdcall;
```

- Parâmetros:

- IDWS do Contas a Receber;

- Possíveis retornos:

- IDWS do Contas a Receber, caso o envio tenha sido bem sucedido;
    - 'ERRO:Contas a Receber não encontrado.', caso o contas a receber não tenha sido encontrado;
    - Descrição do Erro, caso aconteça algum erro durante o envio para contabilidade;

- **CancelaVencimentosDiversosEmAbertoIDWS**

- Assinatura:

```
function CancelaVencimentosDiversosEmAbertoIDWS
(CancelaVencimento: TCancelaVencimento): String; stdcall;
```

- Parâmetros:

- Objeto do tipo TCancelaVencimento;

- Possíveis retornos:

- 'OK', caso o cancelamento tenha sido bem sucedido;
    - 'Cancelamento do Vencimento do Contas a Receber não registrado.', caso o cancelamento do vencimento do contas a receber não tenha sido encontrado;
    - Descrição do Erro, caso aconteça algum erro durante o cancelamento;

- **CancelaVencimentosCartaoaReceberEmAbertoIDWS**

- Assinatura:

```
function CancelaVencimentosCartaoaReceberEmAbertoIDWS
(CancelaVencimento: TCancelaVencimento): String; stdcall;
```

- Parâmetros:

- Objeto do tipo TCancelaVencimento;

- Possíveis retornos:

- 'OK', caso o cancelamento tenha sido bem sucedido;
    - 'Cancelamento do Vencimento do Contas a Receber não registrado.', caso o cancelamento do vencimento do contas a receber não tenha sido encontrado;
    - Descrição do Erro, caso aconteça algum erro durante o cancelamento;

- **CancelaVencimentosChequeaReceberEmAbertoIDWS**

- Assinatura:

```
function CancelaVencimentosChequeaReceberEmAbertoIDWS
(CancelaVencimento: TCancelaVencimento): String; stdcall;
```

- Parâmetros:

- Objeto do tipo TCancelaVencimento;

- Possíveis retornos:

- 'OK', caso o cancelamento tenha sido bem sucedido;
    - 'Cancelamento do Vencimento de Cheque do Contas a Receber não registrado.', caso o cancelamento do vencimento de cheque do contas a receber não tenha sido encontrado;
    - Descrição do Erro, caso aconteça algum erro durante o cancelamento;

- **ExcluiVencimentoDiverso**

- Assinatura:
 

```
function ExcluiVencimentoDiverso(const CodContasReceber: String;
Sequencial: Integer): String; stdcall;
```
  - Parâmetros:
    - Código do contas a receber;
    - Sequencial;
  - Possíveis retornos:
    - 'OK', caso a exclusão tenha sido bem sucedida;
    - 'Vencimento não cadastrado.', caso o vencimento não tenha sido encontrado;
    - Descrição do Erro, caso aconteça algum erro durante a exclusão;
- **getObjetoContasaReceber**
  - Assinatura:
 

```
function getObjetoContasaReceber(const CodContasaReceber:
String): TContasaReceber; stdcall;
```
  - Parâmetros:
    - Código do contas a receber;
  - Possíveis retornos:
    - *nil*, caso não exista contas a receber com o código passado;
    - Objeto do tipo TContasaReceber;
    - Descrição do Erro, caso aconteça algum erro durante a exclusão;
- **getObjetoContasaReceberIDWS**
  - Assinatura:
 

```
function getObjetoContasaReceberIDWS(const IDWS: String):
TContasaReceber; stdcall;
```
  - Parâmetros:
    - IDWS do contas a receber;
  - Possíveis retornos:
    - *nil*, caso não exista contas a receber com o IDWS passado;
    - Objeto do tipo TContasaReceber;
    - Descrição do Erro, caso aconteça algum erro durante a exclusão;
- **GetContasaReceberWithDocumento**
  - Assinatura:
 

```
function GetContasaReceberWithDocumento(const aDocumento:
String): String; stdcall;
```
  - Parâmetros:
    - Documento do contas a receber;
  - Possíveis retornos:
    - String vazia (""), caso não exista contas a receber com o documento passado;
    - Código do contas a receber;
    - Descrição do Erro, caso aconteça algum erro durante a exclusão;
- **GetContasaReceberWithBoleto**
  - Assinatura:
 

```
function GetContasaReceberWithBoleto(const aBoleto: String; const
aIdBloquete: Integer = 1): String; stdcall;
```
  - Parâmetros:

- Número do Boleto;
  - Identificador do Bloquete;
- Possíveis retornos:
  - String vazia (""), caso não exista contas a receber com o documento passado;
  - Código do contas a receber;
  - Descrição do Erro, caso aconteça algum erro durante a exclusão;
- **GetNFPDFStreamDeContasARceber**
  - Assinatura:
 

```
function GetNFPDFStreamDeContasARceber(Emp_Codigo, Cre_Codigo: string): TDadosPDFNotaFiscal; stdcall;
```
  - Parâmetros:
    - Código da empresa;
    - Código do Contas a receber;
  - Possíveis retornos:
    - Objeto do tipo TdadosPDFNotaFiscal. Caso ocorra algum erro, a descrição será retornada na propriedade "MensagemErro" do objeto;
- **GetValorEmAbertoVencimentosDiversosCRE**
  - Assinatura:
 

```
function GetValorEmAbertoVencimentosDiversosCRE(const Codigo_ContasARceber: string; const Sequencial_Vencimento: Integer): Double; stdcall;
```
  - Parâmetros:
    - Código do contas a receber;
    - Sequencial do vencimento;
  - Possíveis retornos:
    - Valor em aberto dos vencimentos diversos do contas a receber, caso o contas a receber passado seja encontrado;
    - -1, caso o contas a receber passado não seja encontrado;
- **GetValorEmAbertoVencimentosCartaoCRE**
  - Assinatura:
 

```
function GetValorEmAbertoVencimentosCartaoCRE(const Codigo_ContasARceber: string; const Sequencial_Vencimento: Integer): Double; stdcall;
```
  - Parâmetros:
    - Código do contas a receber;
    - Sequencial do vencimento;
  - Possíveis retornos:
    - Valor em aberto dos vencimentos de cartão do contas a receber, caso o contas a receber passado seja encontrado;
    - -1, caso o contas a receber passado não seja encontrado;
- **GetValorEmAbertoVencimentosChequeCRE**
  - Assinatura:
 

```
function GetValorEmAbertoVencimentosChequeCRE(const Codigo_ContasARceber: string; const Sequencial_Vencimento: Integer): Double; stdcall;
```

- Parâmetros:
  - Código do contas a receber;
  - Sequencial do vencimento;
- Possíveis retornos:
  - Valor em aberto dos vencimentos de cheque do contas a receber, caso o contas a receber passado seja encontrado;
  - -1, caso o contas a receber passado não seja encontrado;
- **GetVencimentosEmAbertoCRE**
  - Assinatura:
 

```
function GetVencimentosEmAbertoCRE(CNPJCPF: String;
Data:TDateTime): String; stdcall;
```
  - Parâmetros:
    - Código do contas a receber;
    - Sequencial do vencimento;
  - Possíveis retornos:
    - String vazia ("), caso não exista vencimentos em aberto para o contas a receber passado;
    - Lista de vencimentos em aberto separado por ponto e vírgula, caso o contas a receber passado seja encontrado;
- **GetMovimentacaoBaixaARceber**
  - Assinatura:
 

```
GetMovimentacaoBaixaARceber(const EST_Codigo: WideString; const
DataInicial: WideString; const DataFinal: WideString; const
Cancelado: Boolean; const TipodeVencimento: Integer):
TListaMovimentacaoBaixaARceber; stdcall;;
```
  - Parâmetros:
    - Estabelecimento (vazio("")) considera todos);
    - Período (data inicial e final no formato DD/MM/YYYY);
    - Cancelado (true) ou não (false);
    - Tipo de Vencimento, no momento implementado somente Diversos = '1'
  - Possíveis retornos:
    - Array vazia ("), caso não exista baixa conforme os parâmetros passados;
    - Lista de baixas efetuadas no Array;

## 7. Transferências entre Contas Financeiras

### 7.1. TTransferenciaEntreContas

| Código     | Tipo       | Descrição                              | Obrigatório | Observação                                                              |
|------------|------------|----------------------------------------|-------------|-------------------------------------------------------------------------|
| Data       | String(10) | Data da transferência                  | SIM         |                                                                         |
| Sequencial | Integer    | Identificador da transferência na data | SIM         | Se não for informado, o sistema vai gerar o sequencial automaticamente. |
| EstOrigem  | String(4)  | Código do estabelecimento de origem    | SIM         |                                                                         |
| EstDestino | String(4)  | Código do estabelecimento de           | SIM         |                                                                         |



|            |            |                                                                                |     |  |
|------------|------------|--------------------------------------------------------------------------------|-----|--|
|            |            | destino                                                                        |     |  |
| ConOrigem  | String(4)  | Código da conta financeira de origem                                           | SIM |  |
| ConDestino | String(4)  | Código da conta financeira de destino                                          | SIM |  |
| Valor      | Double     | Valor da transferência                                                         | SIM |  |
| Historico  | Blob       | Texto contendo o histórico da transferência                                    | NÃO |  |
| HisCodigo  | String(5)  | Código do histórico                                                            | NÃO |  |
| UsuCodigo  | String(10) | Código do usuário que realizou a transferência                                 | NÃO |  |
| ExportaAC  | Integer    | Indica se deve exportar o Lançamento para o AC Contábil:<br>0 - NÃO<br>1 - SIM | NÃO |  |
| IDWS       | String(30) | Identificador da transferência entre contas incluído via WebService            | NÃO |  |

## 7.2. Métodos

- **IncluiTransferenciaEntreContas**

- Assinatura:

```
function IncluiTransferenciaEntreContas(const
TransferenciaEntreContas: TTransferenciaEntreContas): string;
stdcall;
```

- Parâmetros:

- Objeto do tipo TTransferenciaEntreContas;

- Possíveis retornos:

- Sequencial da transferência entre contas, caso a inclusão tenha sido bem sucedida;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **AlteraTransferenciaEntreContas**

- Assinatura:

```
function AlteraTransferenciaEntreContas(const
TransferenciaEntreContas: TTransferenciaEntreContas): string;
stdcall;
```

- Parâmetros:

- Objeto do tipo TTransferenciaEntreContas;

- Possíveis retornos:

- Sequencial da transferência entre contas, caso a alteração tenha sido bem sucedida;
    - “Transferência não cadastrada”, caso a Transferência passada não seja encontrada;
    - Descrição do Erro, caso aconteça algum erro durante a alteração;

- **ExcluiTransferenciaEntreContas**

- Assinatura:

```
function ExcluiTransferenciaEntreContas(const
TransferenciaEntreContas: TTransferenciaEntreContas): string; stdcall;
```

- Parâmetros:

- Objeto do tipo TTransferenciaEntreContas;

- Possíveis retornos:
  - 'OK', caso a exclusão tenha sido bem sucedida;
  - "Transferência não cadastrada", caso a Transferência passada não seja encontrada;
  - Descrição do Erro, caso aconteça algum erro durante a exclusão;

- **GetTransferenciaEntreContas**

- Assinatura:

```
function GetTransferenciaEntreContas(const Data: string; Sequencial: Integer): TTransferenciaEntreContas; stdcall;
```

- Parâmetros:

- Data;
- Sequencial;

- Possíveis retornos:

- *nil*, caso não seja encontrada transferência com as informações passadas;
- Objeto do tipo TtransferenciaEntreContas, caso seja encontrada transferência com as informações passadas;

- **GetTransferenciaEntreContasIDWS**

- Assinatura:

```
function GetTransferenciaEntreContasIDWS(const IDWS: String): TTransferenciaEntreContas; stdcall;
```

- Parâmetros:

- Data;
- Sequencial;

- Possíveis retornos:

- *nil*, caso não seja encontrada transferência com as informações passadas;
- Objeto do tipo TtransferenciaEntreContas, caso seja encontrada transferência com as informações passadas;

## 8. Adiantamento a clientes

### 8.1. TAdiantamentoCliente

| Campo               | Tipo       | Descrição                                                   | Obrigatório | Observação          |
|---------------------|------------|-------------------------------------------------------------|-------------|---------------------|
| Estabelecimento     | String(4)  | Código do estabelecimento                                   | SIM         |                     |
| ContaFinanceira     | String(4)  | Código da conta financeira                                  | SIM         |                     |
| Data                | XSDatetime |                                                             | SIM         |                     |
| CentroResultados    | String(10) | Código do centro de resultados                              | SIM         |                     |
| ReceitaAdiantamento | String(10) | Código da receita do adiantamento                           | SIM         |                     |
| Cliente             | String(6)  |                                                             | SIM         |                     |
| Valor               | Double     |                                                             | SIM         |                     |
| Observacao          | Blob       |                                                             | NÃO         |                     |
| ExportaAC           | Integer    | Indica se o movimento deve ser exportado para o AC Contábil | SIM         | 0 - Não;<br>1 - Sim |
| ClienteCNPJCPF      | String(14) |                                                             | NÃO         |                     |
| ClienteIDWS         | String(30) | Identificador do cliente incluído via WebService            | NÃO         |                     |

|      |            |                                                                 |     |  |
|------|------------|-----------------------------------------------------------------|-----|--|
| IDWS | String(30) | Identificador do adiantamento a cliente incluído via WebService | NÃO |  |
|------|------------|-----------------------------------------------------------------|-----|--|

## 8.2. Métodos

- **IncluiAdiantamentoCliente**

- Assinatura:

```
function IncluiAdiantamentoCliente(const AdiantamentoCliente: TAdiantamentoCliente): String; stdcall;
```

- Parâmetros:

- Objeto do tipo TAdiantamentoCliente;

- Possíveis retornos:

- Sequencial da transferência entre contas, caso a inclusão tenha sido bem sucedida;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **AlteraTransferenciaEntreContas**

- Assinatura:

```
function AlteraTransferenciaEntreContas(const TransferenciaEntreContas: TTransferenciaEntreContas): string; stdcall;
```

- Parâmetros:

- Objeto do tipo TTransferenciaEntreContas;

- Possíveis retornos:

- Sequencial da transferência entre contas, caso a alteração tenha sido bem sucedida;
    - “Transferência não cadastrada”, caso a Transferência passada não seja encontrada;
    - Descrição do Erro, caso aconteça algum erro durante a alteração;

- **ExcluiTransferenciaEntreContas**

- Assinatura:

```
function ExcluiTransferenciaEntreContas(const TransferenciaEntreContas: TTransferenciaEntreContas): string; stdcall;
```

- Parâmetros:

- Objeto do tipo TTransferenciaEntreContas;

- Possíveis retornos:

- ‘OK’, caso a exclusão tenha sido bem sucedida;
    - “Transferência não cadastrada”, caso a Transferência passada não seja encontrada;
    - Descrição do Erro, caso aconteça algum erro durante a exclusão;

- **AlteraAdiantamentoCliente**

- Assinatura:

```
function AlteraAdiantamentoCliente(const AdiantamentoCliente: TAdiantamentoCliente): String; stdcall;
```

- Parâmetros:

- Objeto do tipo TAdiantamentoCliente;

- Possíveis retornos:

- Código do Adiantamento ao Cliente, caso a alteração tenha sido bem sucedida;
    - “Adiantamento de Cliente não cadastrado.(AlteraAdiantamentoCliente)”, caso a Transferência passada não seja encontrada;

- Descrição do Erro, caso aconteça algum erro durante a alteração;
- **ExcluiAdiantamentoCliente**
  - Assinatura:
 

```
function ExcluiAdiantamentoCliente(const Codigo: string): string;
stdcall;
```
  - Parâmetros:
    - Código do Adiantamento ao Cliente;
  - Possíveis retornos:
    - 'OK', caso a exclusão tenha sido bem sucedida;
    - "Adiantamento a Cliente não cadastrado.", caso o Adiantamento passado não seja encontrado;
    - Descrição do Erro, caso aconteça algum erro durante a exclusão;
- **GetObjAdiantamentoCliente**
  - Assinatura:
 

```
function GetObjAdiantamentoCliente(const Codigo: string):
TAdiantamentoCliente; stdcall;
```
  - Parâmetros:
    - Código do Adiantamento a Cliente;
  - Possíveis retornos:
    - Objeto do tipo TAdiantamentoCliente, caso o Adiantamento passado seja encontrado;
    - "Adiantamento a Cliente não encontrado.", caso o Adiantamento passado não seja encontrado;
    - Descrição do Erro, caso aconteça algum erro durante a busca;
- **GetObjAdiantamentoClienteIDWS**
  - Assinatura:
 

```
function GetObjAdiantamentoClienteIDWS(const IDWS: string):
TAdiantamentoCliente; stdcall;
```
  - Parâmetros:
    - IDWS do Adiantamento a Cliente;
  - Possíveis retornos:
    - Objeto do tipo TAdiantamentoCliente, caso exista Adiantamento com o IDWS passado;
    - "Adiantamento a Cliente não encontrado.", caso não exista Adiantamento com o IDWS passado;
    - Descrição do Erro, caso aconteça algum erro durante a busca;
- **CancelarAdiantamentoCliente**
  - Assinatura:
 

```
function CancelarAdiantamentoCliente(CodigoACL: String;
DataCancelamento: TDateTime; MotivoCancelamento: String): String;
stdcall;
```
  - Parâmetros:
    - Código do Adiantamento a Cliente;
    - Data do Cancelamento;
    - Motivo do Cancelamento;

- Possíveis retornos:
  - 'OK', caso o cancelamento tenha sido bem sucedido;
  - "Adiantamento de Cliente não cadastrado.(CancelarAdiantamentoCliente).", caso o Adiantamento passado não seja encontrado;
  - Descrição do Erro, caso aconteça algum erro durante o cancelamento;
- **IncluiArrayAdiantamentoCliente**
  - Assinatura:
 

```
function IncluiArrayAdiantamentoCliente (ArrayAdiantamentoCliente: TArrayAdiantamentoCliente): TArrayAdiantamentoCliente; stdcall;
```
  - Parâmetros:
    - Array de Adiantamento a Cliente;
  - Possíveis retornos:
    - Array de Adiantamento a Cliente, caso a inclusão tenha sido bem sucedida;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão;
- **AlteraArrayAdiantamentoCliente**
  - Assinatura:
 

```
function AlteraArrayAdiantamentoCliente (ArrayAdiantamentoCliente: TArrayAdiantamentoCliente): TArrayAdiantamentoCliente; stdcall;
```
  - Parâmetros:
    - Array de Adiantamento a Cliente;
  - Possíveis retornos:
    - Array de Adiantamento a Cliente, caso a alteração tenha sido bem sucedida;
    - Descrição do Erro, caso aconteça algum erro durante a alteração;
- **ExcluiArrayAdiantamentoCliente**
  - Assinatura:
 

```
function ExcluiArrayAdiantamentoCliente (ArrayAdiantamentoCliente: TArrayAdiantamentoCliente): Boolean; stdcall;
```
  - Parâmetros:
    - Array de Adiantamento a Cliente;
  - Possíveis retornos:
    - "True", caso a exclusão tenha sido bem sucedida;
    - "False", caso a exclusão não tenha sido bem sucedida;
    - Descrição do Erro, caso aconteça algum erro durante a exclusão;
- **CancelarArrayAdiantamentoCliente**
  - Assinatura:
 

```
function CancelarArrayAdiantamentoCliente
 ArrayCancelaAdiantamentoCliente:
 TarrayCancelaAdiantamentoCliente):
 TArrayCancelaAdiantamentoCliente; stdcall;
```
  - Parâmetros:
    - Array de Adiantamento a Cliente a serem cancelados;
  - Possíveis retornos:
    - Array de Adiantamentos a Cliente cancelados, caso o cancelamento tenha sido bem sucedido;

- Descrição do Erro, caso aconteça algum erro durante a exclusão;

## 9. Notas Fiscais de Saída

### 9.1. TNotasFiscaisSaida

| Campo                | Tipo       | Descrição                                            | Obrigatório | Observação                                                          |
|----------------------|------------|------------------------------------------------------|-------------|---------------------------------------------------------------------|
| Estabelecimento      | String(4)  | Código do estabelecimento                            | SIM         |                                                                     |
| Cliente              | String(6)  | Código do cliente                                    | SIM         |                                                                     |
| Formulario           | String(10) | Número do Formulário                                 | NÃO         |                                                                     |
| Numero               | String(10) | Número da Nota Fiscal                                | NÃO         |                                                                     |
| Serie                | String(3)  | Série da Nota Fiscal                                 | NÃO         |                                                                     |
| Subserie             | String(3)  |                                                      | NÃO         |                                                                     |
| AIDF                 | String(25) |                                                      | NÃO         |                                                                     |
| AMIDF                | String(10) |                                                      | NÃO         |                                                                     |
| DataEmissao          | String(10) |                                                      | NÃO         |                                                                     |
| DataSaida            | String(10) |                                                      | NÃO         |                                                                     |
| CFOP                 | String(4)  |                                                      | NÃO         |                                                                     |
| Cancelada            | Integer    | Indica se a nota fiscal está cancelada               | NÃO         | 0 - Não;<br>1 - Sim                                                 |
| ValorFrete           | Double     |                                                      | NÃO         |                                                                     |
| ValorSeguro          | Double     |                                                      | NÃO         |                                                                     |
| ValorOutrasDespesas  | Double     |                                                      | NÃO         |                                                                     |
| ValorIPI             | Double     |                                                      | NÃO         |                                                                     |
| BaselCMS             | Double     |                                                      | NÃO         |                                                                     |
| ValorICMS            | Double     |                                                      | NÃO         |                                                                     |
| ValorProdutos        | Double     |                                                      | NÃO         |                                                                     |
| ValorTotal           | Double     |                                                      | NÃO         |                                                                     |
| Observacao           | Blob       |                                                      | NÃO         |                                                                     |
| ObsCorpoNF           | Blob       |                                                      | NÃO         |                                                                     |
| ICMSSobreIPI         | Integer    |                                                      | NÃO         |                                                                     |
| SeloNumero           | String(10) |                                                      | NÃO         |                                                                     |
| SeloSerie            | String(2)  |                                                      | NÃO         |                                                                     |
| Especie              | String(5)  | Espécie da Nota Fiscal                               | SIM         | NFE - Notas Fiscais de Produtos;<br>NFSE - Notas Fiscais de Serviço |
| BaselCMSSubst        | Double     |                                                      | NÃO         |                                                                     |
| ValorICMSSubst       | Double     |                                                      | NÃO         |                                                                     |
| NaturezaOperacao     | String(4)  | Código da tabela de Natureza da Operação             | NÃO         |                                                                     |
| ChaveNFe             | String(47) | Chave Eletrônica da Nota Fiscal                      | NÃO         |                                                                     |
| CRE_CODIGO           | String(11) | Código do Contas a Receber                           | NÃO         |                                                                     |
| IDWS                 | String(30) | Identificador da Nota Fiscal incluído via WebService | NÃO         |                                                                     |
| ClienteIDWS          | String(30) | Identificador do cliente incluído via WebService     | NÃO         |                                                                     |
| CLI_CNPJCPF          | String(14) | CNPJ\CPF do cliente                                  | NÃO         |                                                                     |
| TipoEmitente         | String(1)  |                                                      | NÃO         |                                                                     |
| ValorServicos        | Double     |                                                      | NÃO         |                                                                     |
| ItensNotaFiscalSaida | Array of   |                                                      | (*)         |                                                                     |

|                         |                              |  |       |  |
|-------------------------|------------------------------|--|-------|--|
|                         | TltesnNotaFiscalSaida        |  |       |  |
| ServicosNotaFiscalSaida | ArrayServicosNotaFiscalSaida |  | (*)   |  |
| DocsRefsNotaFiscalSaida | ArrayDocsRefsNotaFiscalSaida |  | (**)  |  |
| FormasPagamentosNFS     | ArrayFormasPagamentosNFS     |  | (***) |  |

(\*) obrigatório ter ItensNotaFiscalSaida OU ServicosNotaFiscalSaida

(\*\*) obrigatório para NFe e Nfce

(\*\*\*) obrigatório para notas de devolução ou nota complementar

### 9.1.1. TltesnNotaFiscalSaida

| Campo             | Tipo       | Descrição                               | Obrigatório | Observação              |
|-------------------|------------|-----------------------------------------|-------------|-------------------------|
| NFS_Codigo        | String(11) | Código da Nota Fiscal                   | SIM         | Somente para informação |
| Sequencial        | Integer    |                                         | SIM         | Gerado automaticamente  |
| Item              | String(10) | Código do Produto                       | SIM         |                         |
| CFOP              | String(4)  |                                         | NÃO         |                         |
| STA_Codigo        | String(1)  |                                         | NÃO         |                         |
| STB_Codigo        | String(2)  |                                         | NÃO         |                         |
| AliquotaICMS      | Double     |                                         | NÃO         |                         |
| AliquotaIPI       | Double     |                                         | NÃO         |                         |
| Unidade           | String(3)  |                                         | NÃO         |                         |
| Valor             | Double     |                                         | NÃO         |                         |
| CST_IPI           | String(2)  | Código da Situação Tributária do IPI    | NÃO         | Tabela 20               |
| CST_PIS           | String(2)  | Código da Situação Tributária do PIS    | NÃO         | Tabela 8                |
| CST_COFINS        | String(2)  | Código da Situação Tributária do COFINS | NÃO         | Tabela 8                |
| Quantidade        | Double     |                                         | NÃO         |                         |
| QtdTributavel     | Double     |                                         | NÃO         |                         |
| VlrUnitTributavel | Double     |                                         | NÃO         |                         |
| UndTributavel     | String(3)  |                                         | NÃO         |                         |

### 9.1.2. TServicosNotaFiscalSaida

| Campo                | Tipo       | Descrição             | Obrigatório | Observação              |
|----------------------|------------|-----------------------|-------------|-------------------------|
| NFS_Codigo           | String(11) | Código da Nota Fiscal | SIM         | Somente para informação |
| Sequencial           | Integer    |                       | SIM         | Gerado automaticamente  |
| CodigoServico        | String(4)  |                       | SIM         |                         |
| DescricaoNotaFiscal  | String(60) |                       | NÃO         |                         |
| CFOP                 | String(4)  |                       | NÃO         |                         |
| AliquotaISS          | Double     |                       | NÃO         |                         |
| Valor                | Double     |                       | NÃO         |                         |
| ComplementoDescricao | Blob       |                       | NÃO         |                         |
| AliquotaICMS         | Double     |                       | NÃO         |                         |
| QtdeServicos         | Double     |                       | NÃO         |                         |

|                               |            |  |     |  |
|-------------------------------|------------|--|-----|--|
| STISS_CODIGO                  | String(2)  |  | NÃO |  |
| DeducaoISS                    | Double     |  | NÃO |  |
| Desconto                      | Double     |  | NÃO |  |
| ItemListaServico              | String(10) |  | NÃO |  |
| CodTributMunicipal            | String(20) |  | NÃO |  |
| UF_Sigla                      | String(2)  |  | NÃO |  |
| CodigoMunicipio               | String(5)  |  | NÃO |  |
| ReducaoICMS                   | Double     |  | NÃO |  |
| CodigoObra                    | String(10) |  | NÃO |  |
| ValorBruto                    | Double     |  | NÃO |  |
| DescontoIncondicional         | Double     |  | NÃO |  |
| PercDescIncondicional         | Double     |  | NÃO |  |
| SitTributCOFINS               | String(2)  |  | NÃO |  |
| SitTributPIS                  | String(2)  |  | NÃO |  |
| AliquotaNacional              | Double     |  | NÃO |  |
| AliquotaImportacao            | Double     |  | NÃO |  |
| AliquotaEstadual              | Double     |  | NÃO |  |
| AliquotaMunicipal             | Double     |  | NÃO |  |
| CodigoModalidadeServico       | String(2)  |  | NÃO |  |
| CodigoNaturezaTributacaoISS   | String(3)  |  | NÃO |  |
| CodigoOperacaoServicoPrestado | String(3)  |  | NÃO |  |
| CodigoExigibilidadeISS        | String(3)  |  | NÃO |  |
| AliquotaINSS                  | Double     |  | NÃO |  |

### 9.1.3. TDocsRefsNotaFiscalSaida

| Campo            | Tipo       | Descrição             | Obrigatório | Observação                                                                                           |
|------------------|------------|-----------------------|-------------|------------------------------------------------------------------------------------------------------|
| NFS_Codigo       | String(11) | Código da Nota Fiscal | SIM         | Somente para informação                                                                              |
| Sequencial       | Integer    |                       | SIM         | Gerado automaticamente                                                                               |
| Tipo             | String(1)  |                       | SIM         | N - Nfe<br>A - NF1/1A<br>P - NF Produtor Rural<br>C - Cupom Fiscal<br>X – Excesso de bagagem CT-E OS |
| NFE_Chave        | String(47) |                       | NÃO         |                                                                                                      |
| UFD_Codigo       | String(2)  |                       | NÃO         |                                                                                                      |
| AnoMes           | String(4)  |                       | NÃO         |                                                                                                      |
| CNPJ             | String(14) |                       | NÃO         |                                                                                                      |
| Modelo           | String(2)  |                       | NÃO         |                                                                                                      |
| Serie            | String(3)  |                       | NÃO         |                                                                                                      |
| Numero           | String(9)  |                       | NÃO         |                                                                                                      |
| CPF              | String(11) |                       | NÃO         |                                                                                                      |
| IE               | String(14) |                       | NÃO         |                                                                                                      |
| Numero_ECF       | String(3)  |                       | NÃO         |                                                                                                      |
| Numero_COO       | String(6)  |                       | NÃO         |                                                                                                      |
| ModeloNF1A       | String(2)  |                       | NÃO         |                                                                                                      |
| UFD_CodigoNFProd | String(2)  |                       | NÃO         |                                                                                                      |
| AnoMesNFProd     | String(4)  |                       | NÃO         |                                                                                                      |
| CNPJNFProd       | String(14) |                       | NÃO         |                                                                                                      |
| ModeloNFProd     | String(2)  |                       | NÃO         |                                                                                                      |



|                   |            |  |     |  |
|-------------------|------------|--|-----|--|
| SerieNFProd       | String(3)  |  | NÃO |  |
| NumeroNFProd      | String(9)  |  | NÃO |  |
| NFE_ChaveNFProd   | String(47) |  | NÃO |  |
| DTEmissao         | XSDatetime |  | NÃO |  |
| NFPropria         | Integer    |  | NÃO |  |
| CLI_Codigo        | String(6)  |  | NÃO |  |
| Subserie          | String(3)  |  | NÃO |  |
| ValorTransportado | Double     |  | NÃO |  |

#### 9.1.4. TFormasPagamentosNFS

| Campo             | Tipo       | Descrição             | Obrigatório | Observação              |
|-------------------|------------|-----------------------|-------------|-------------------------|
| NFS_Codigo        | String(11) | Código da Nota Fiscal | SIM         | Somente para informação |
| Sequencial        | Integer    |                       | SIM         | Gerado automaticamente  |
| FormaDePagamento  | String(2)  |                       | SIM         |                         |
| Bandeira          | String(2)  |                       | NÃO         |                         |
| NumeroAutorizacao | String(20) |                       | NÃO         |                         |
| CNPJCredenciadora | String(14) |                       | NÃO         |                         |
| ValorPagamento    | Double     |                       | NÃO         |                         |
| TipoIntegracao    | String(1)  |                       | NÃO         |                         |

## 9.2. Métodos

- **IncluirNotaFiscalSaida**

- Assinatura:
 

```
function IncluirNotaFiscalSaida(NotaFiscal: TNotasFiscaisSaida): String; stdcall;
```

- Parâmetros:

- Objeto do tipo TNotasFiscaisSaida;

- Possíveis retornos:

- Código da Nota Fiscal de Saída, caso a inclusão tenha sido bem sucedida;
- Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **ExcluirNotaFiscalSaida**

- Assinatura:
 

```
function ExcluirNotaFiscalSaida(const Codigo: String): String; stdcall;
```

- Parâmetros:

- Código da Nota Fiscal de Saída;

- Possíveis retornos:

- 'OK', caso a exclusão tenha sido bem sucedida;
- 'ERRO: Nota Fiscal não encontrada. Não foi possível excluir', caso a Nota Fiscal passada não tenha sido encontrada;
- Descrição do Erro, caso aconteça algum erro durante a exclusão;

- **GetNotaFiscalSaida**

- Assinatura:

```
function GetNotaFiscalSaida(const Codigo: String):
TNotasFiscaisSaida; stdcall;
```

- Parâmetros:
  - Código da Nota Fiscal de Saída;
- Possíveis retornos:
  - Objeto do tipo TNotasFiscaisSaida, caso exista nota fiscal de saída com o código passado;
  - *nil* e mensagem 'Nota Fiscal não encontrada.', caso não exista nota fiscal de saída com o código passado ;
  - Descrição do Erro, caso aconteça algum erro durante a busca;

- **GetNotaFiscalSaidaIDWS**

```
Assinatura:
function GetNotaFiscalSaidaIDWS(const IDWS: String):
TNotasFiscaisSaida; stdcall;
```

- Parâmetros:
  - IDWS da Nota Fiscal de Saída;
- Possíveis retornos:
  - Objeto do tipo TNotasFiscaisSaida, caso exista nota fiscal de saída com o IDWS passado;
  - *nil* e mensagem 'Nota Fiscal não encontrada.', caso não exista nota fiscal de saída com o IDWS passado;
  - Descrição do Erro, caso aconteça algum erro durante a busca;

- **GetNotaFiscalSaidaCRE**

```
Assinatura:
function GetNotaFiscalSaidaCRE(const CRE: String):
TNotasFiscaisSaida; stdcall;
```

- Parâmetros:
  - Código do Contas a Receber;
- Possíveis retornos:
  - Objeto do tipo TNotasFiscaisSaida, caso exista nota fiscal de saída com o Código do Contas a Receber passado;
  - *nil* e mensagem 'Nota Fiscal não encontrada.', caso não exista nota fiscal de saída com o Código do Contas a Receber passado;
  - Descrição do Erro, caso aconteça algum erro durante a busca;

## 10. Contas a Pagar

### 10.1. TContasaPagar

| Campo            | Tipo       | Descrição                      | Obrigatório | Observação             |
|------------------|------------|--------------------------------|-------------|------------------------|
| Codigo           | String(11) |                                | SIM         | Gerado automaticamente |
| Estabelecimento  | String(4)  | Código do estabelecimento      | SIM         |                        |
| CentroResultados | String(10) | Código do Centro de Resultados | NÃO         |                        |
| Fornecedor       | String(14) | CNPJ do Fornecedor             | (*)         |                        |
| CodigoFornecedor | String(6)  | Código do Fornecedor           | (*)         |                        |
| Despesa          | String(10) | Código da Despesa              | NÃO         |                        |
| Documento        | String(40) |                                | NÃO         |                        |
| TipoDocumento    | String(3)  | Código do tipo de              | SIM         |                        |

|                   |                              |                                                                                   |     |                                                           |
|-------------------|------------------------------|-----------------------------------------------------------------------------------|-----|-----------------------------------------------------------|
|                   |                              | documento                                                                         |     |                                                           |
| Emissao           | String(10)                   | Data de emissão                                                                   | SIM | 01/10/2015                                                |
| MesAnoComp        | String(6)                    | Mês e ano de competência                                                          | NÃO | 201510                                                    |
| DtEntrada         | String(10)                   | Data de entrada                                                                   | SIM | 01/10/2015                                                |
| Valor             | Double                       |                                                                                   | NÃO |                                                           |
| ContaFinanceira   | String(4)                    | Código da conta financeira                                                        | NÃO |                                                           |
| ExportaAC         | Integer                      | Indica se deve exportar o lançamento para o AC<br>Contábil:<br>0 - NÃO<br>1 - SIM | NÃO | Se não informar será incluído automaticamente o valor '0' |
| OBS               | Blob                         |                                                                                   | NÃO |                                                           |
| ISSValor          | Double                       |                                                                                   | NÃO |                                                           |
| ISSBase           | Double                       |                                                                                   | NÃO |                                                           |
| IRRFBase          | Double                       |                                                                                   | NÃO |                                                           |
| IRRFValorCPF      | Double                       |                                                                                   | NÃO |                                                           |
| IRRFValorCNPJ     | Double                       |                                                                                   | NÃO |                                                           |
| INSSValorCPF      | Double                       |                                                                                   | NÃO |                                                           |
| INSSValorCNPJ     | Double                       |                                                                                   | NÃO |                                                           |
| INSSBase          | Double                       |                                                                                   | NÃO |                                                           |
| PISCOFINSCSLValor | Double                       |                                                                                   | NÃO |                                                           |
| PISCOFINSCSLBase  | Double                       |                                                                                   | NÃO |                                                           |
| Origem            | String(1)                    |                                                                                   | NÃO | Tabela 17 - Origem do Contas a Pagar                      |
| VencimentosaPagar | Array of TVencimentosa Pagar |                                                                                   | SIM |                                                           |
| DespesaPagar      | TArrayOfDespesaPagar         |                                                                                   | SIM |                                                           |
| IDWS              | String(30)                   | Indicador do Web Service                                                          | NÃO |                                                           |
| FornecedorIDWS    | String(30)                   | Indicador do Web Service do Fornecedor                                            | NÃO |                                                           |
| ContaAplicacao    | String                       |                                                                                   | NÃO |                                                           |
| FormatoData       | String                       | Formatação da Data                                                                | NÃO |                                                           |
| EmissaoNF         | String(10)                   |                                                                                   | NÃO | 01/10/2015                                                |

(\*) Obrigatório informar o CNPJ ou o Código do Fornecedor

### 10.1.1. TVencimentosaPagar

| Campo      | Tipo       | Descrição                 | Obrigatório | Observação              |
|------------|------------|---------------------------|-------------|-------------------------|
| Codigo     | String(11) |                           | SIM         | Somente para informação |
| Sequencial | Integer    |                           | SIM         | Gerado automaticamente  |
| Vencimento | String(10) | Data do vencimento        | NÃO         |                         |
| Valor      | Double     |                           | NÃO         |                         |
| Titulo     | String(20) | Número original do título | NÃO         |                         |
| DtCancel   | String(10) |                           | NÃO         |                         |

### 10.1.2. TBaixaVencimentoaPagar

| Campo                  | Tipo       | Descrição                                                                   | Obrigatório | Observação              |
|------------------------|------------|-----------------------------------------------------------------------------|-------------|-------------------------|
| Codigo                 | String(11) |                                                                             | SIM         | Somente para informação |
| SequencialVencimento   | Integer    |                                                                             | SIM         | Gerado automaticamente  |
| DataVencimento         | String(10) |                                                                             | NÃO         |                         |
| Sequencial             | Integer    |                                                                             | NÃO         |                         |
| Data                   | String(10) |                                                                             | NÃO         |                         |
| Valor                  | Double     |                                                                             | NÃO         |                         |
| Desconto               | Double     |                                                                             | NÃO         |                         |
| Juros                  | Double     |                                                                             | NÃO         |                         |
| DtCancel               | String(10) |                                                                             | NÃO         |                         |
| MotivoCancel           | Blob       |                                                                             | NÃO         |                         |
| Historico              | String()   |                                                                             | NÃO         |                         |
| LAN_ContaFinanceira    | String(4)  |                                                                             | NÃO         |                         |
| LAN_Historico          | Blob       |                                                                             | NÃO         |                         |
| LAN_CRD_Desconto       | String(10) |                                                                             | NÃO         |                         |
| LAN_CRD_Juros          | String(10) |                                                                             | NÃO         |                         |
| LAN_REFERENCIABANCARIA | String()   |                                                                             | NÃO         |                         |
| IDWS                   | String(30) | Identificador da baixa de vencimento a pagar incluído via <i>WebService</i> | NÃO         |                         |
| CPGIDWS                | String(30) | Identificador do contas a pagar incluído via <i>WebService</i>              | NÃO         |                         |
| FormatoData            | String()   |                                                                             | NÃO         |                         |

### 10.1.3. TDespesaPagar

| Campo            | Tipo       | Descrição | Obrigatório | Observação              |
|------------------|------------|-----------|-------------|-------------------------|
| Codigo           | String(11) |           | SIM         | Somente para informação |
| Estabelecimento  | String(4)  |           | SIM         | Gerado automaticamente  |
| CentroResultados | String(10) |           | SIM         |                         |
| Despesa          | String(10) |           | SIM         |                         |
| Valor            | Double     |           | SIM         |                         |
| Obs              | Blob       |           | NÃO         |                         |

## 10.2. Métodos

- **IncluiContasaPagar**
  - Assinatura:  

```
function IncluiContasaPagar(const ContasaPagar: TContasaPagar): string; stdcall;
```
  - Parâmetros:
    - Objeto do tipo TContasaPagar;
  - Possíveis retornos:
    - Código do Contas a Pagar, caso a inclusão tenha sido bem sucedida;

- Descrição do Erro, caso aconteça algum erro durante a inclusão;
- **ExcluiContasaPagar**
  - Assinatura:
 

```
function ExcluiContasaPagar(const Codigo: string): string;
stdcall;
```
  - Parâmetros:
    - Código do Contas a Pagar;
  - Possíveis retornos:
    - 'OK', caso a exclusão tenha sido bem sucedida;
    - 'Contas a Pagar não cadastrado.(ExcluiContasaPagar)', caso o contas a pagar passado não seja encontrado;
    - Descrição do Erro, caso aconteça algum erro durante a exclusão;
- **GetObjContasaPagar**
  - Assinatura:
 

```
function GetObjContasaPagar(const Codigo: string):
TContasaPagar; stdcall;
```
  - Parâmetros:
    - Código do Contas a Pagar;
  - Possíveis retornos:
    - Objeto do tipo TContasaPagar, caso a busca tenha sido bem sucedida;
    - 'Contas a Pagar não encontrado.', caso o contas a pagar passado não seja encontrado;
    - Descrição do Erro, caso aconteça algum erro durante a busca;
- **GetObjContasaPagarIDWS**
  - Assinatura:
 

```
function GetObjContasaPagarIDWS(const IDWS: string):
TContasaPagar; stdcall;
```
  - Parâmetros:
    - IDWS do Contas a Pagar;
  - Possíveis retornos:
    - Objeto do tipo TContasaPagar, caso a busca tenha sido bem sucedida;
    - 'Contas a Pagar não encontrado.', caso o contas a pagar passado não seja encontrado;
    - Descrição do Erro, caso aconteça algum erro durante a busca;
- **IncluirBaixaContasaPagar**
  - Assinatura:
 

```
function IncluirBaixaContasaPagar(const BaixaContasaPagar:
TBaixaVencimentoaPagar): string; stdcall;
```
  - Parâmetros:
    - Objeto do tipo TBaixaVencimentoaPagar;
  - Possíveis retornos:
    - 'OK', caso a busca tenha sido bem sucedida;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão;
- **CancelarBaixaContasaPagar**
  - Assinatura:
 

```
function CancelarBaixaContasaPagar(CodigoCPG: string;
SequencialVencimento, Sequencial: Integer; DataVencimento,
```

```
DataCancelamento: string; MotivoCancelamento: string): string;
stdcall;
```

- **Parâmetros:**
  - Código do Contas a Pagar;
  - Sequencial do Vencimento;
  - Sequencial da Baixa;
  - Data do Vencimento;
  - Data do Cancelamento;
  - Motivo do Cancelamento;
- **Possíveis retornos:**
  - 'OK', caso o cancelamento da baixa tenha sido bem sucedida;
  - 'Baixa do Vencimento do Contas a Pagar não cadastrada', caso a baixa não tenha sido encontrada;
  - Descrição do Erro, caso aconteça algum erro durante o cancelamento;

- **CancelaBaixaContasaPagarIDWS**

- **Assinatura:**

```
function CancelaBaixaContasaPagarIDWS (CancelaBaixaVencimento:
TCancelaBaixaVencimento): string; stdcall;
```

- **Parâmetros:**
  - Objeto do tipo TCancelaBaixaVencimento;
- **Possíveis retornos:**
  - 'OK', caso o cancelamento da baixa tenha sido bem sucedida;
  - 'Baixa do Vencimento do Contas a Pagar não cadastrada', caso a baixa não tenha sido encontrada;
  - Descrição do Erro, caso aconteça algum erro durante o cancelamento;

- **CancelaVencimentosaPagarEmAberto**

- **Assinatura:**

```
function CancelaVencimentosaPagarEmAberto (const CodContasaPagar:
string; sDtCancel: TDateTime; aMotCancel: string): string;
stdcall;
```

- **Parâmetros:**
  - Código do Contas a Pagar;
  - Data do cancelamento;
  - Motivo do cancelamento;
- **Possíveis retornos:**
  - 'OK', caso o cancelamento do contas a pagar tenha sido bem sucedido;
  - 'Contas a Pagar não cadastrado', caso o contas a pagar não tenha sido encontrado;
  - Descrição do Erro, caso aconteça algum erro durante o cancelamento;

- **CancelaVencimentosaPagarEmAbertoIDWS**

- **Assinatura:**

```
function CancelaVencimentosaPagarEmAbertoIDWS (CancelaVencimento:
TCancelaVencimento): string; stdcall;
```
- **Parâmetros:**

- Objeto do Tipo TCancelaVencimento;
- Possíveis retornos:
  - 'OK', caso o cancelamento do contas a pagar tenha sido bem sucedido;
  - 'Contas a Pagar não cadastrado', caso o contas a pagar não tenha sido encontrado;
  - Descrição do Erro, caso aconteça algum erro durante o cancelamento;

### 11.1. TAdiantamentoFornecedor

| Campo               | Tipo       | Descrição                                                                                     | Obrigatório | Observação |
|---------------------|------------|-----------------------------------------------------------------------------------------------|-------------|------------|
| Codigo              | String(11) |                                                                                               | SIM         |            |
| Estabelecimento     | String(4)  |                                                                                               | SIM         |            |
| ContaFinanceira     | String(4)  |                                                                                               | SIM         |            |
| Data                | XSDatetime |                                                                                               | SIM         |            |
| CentroResultados    | String(10) |                                                                                               | SIM         |            |
| DespesaAdiantamento | String(11) |                                                                                               | SIM         |            |
| Fornecedor          | String(6)  |                                                                                               | SIM         |            |
| FornecedorCNPJCPF   | String(14) |                                                                                               | NÃO         |            |
| Valor               | Double     |                                                                                               | SIM         |            |
| Observacao          | Blob       |                                                                                               | NÃO         |            |
| ExportaAC           | Integer    | Indica se deve exportar o Adiantamento a Fornecedor para o AC Contábil:<br>0 - NÃO<br>1 - SIM | SIM         |            |

### 11.2. Métodos

- **IncluiAdiantamentoFornecedor**

- Assinatura:

```
function IncluiAdiantamentoFornecedor(const
AdiantamentoFornecedor: TAdiantamentoFornecedor): string;
stdcall;
```

- Parâmetros:

- Objeto do tipo TAdiantamentoFornecedor;

- Possíveis retornos:

- Código do Adiantamento a Fornecedor, caso a inclusão tenha sido bem sucedida;
- Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **AlteraAdiantamentoFornecedor**

- Assinatura:

```
function AlteraAdiantamentoFornecedor(const
AdiantamentoFornecedor: TAdiantamentoFornecedor): string;
stdcall;
```

- Parâmetros:

- Objeto do tipo TAdiantamentoFornecedor;

- Possíveis retornos:

- Código do Adiantamento a Fornecedor, caso a alteração tenha sido bem sucedida;
- 'Adiantamento de Fornecedor não cadastrado.(AlteraAdiantamentoFornecedor)' caso o adiantamento não tenha sido encontrado;
- Descrição do Erro, caso aconteça algum erro durante a alteração;

- **ExcluiAdiantamentoFornecedor**

- Assinatura:

```
function ExcluiAdiantamentoFornecedor(const Codigo: string):
string; stdcall;
```

- Parâmetros:

- Código do Adiantamento a Fornecedor;

- Possíveis retornos:

- 'OK', caso a exclusão tenha sido bem sucedida;
    - 'Adiantamento de Fornecedor não cadastrado.', caso o adiantamento não tenha sido encontrado;
    - Descrição do Erro, caso aconteça algum erro durante a exclusão;

- **GetObjAdiantamentoFornecedor**

- Assinatura:

```
function GetObjAdiantamentoFornecedor(const Codigo: string):
TAdiantamentoFornecedor; stdcall;
```

- Parâmetros:

- Código do Adiantamento a Fornecedor;

- Possíveis retornos:

- Objeto do tipo TAdiantamentoFornecedor, caso o adiantamento a fornecedor tenha sido encontrado;
    - *nil*, caso o adiantamento a fornecedor não tenha sido encontrado;
    - Descrição do Erro, caso aconteça algum erro durante a busca;

## 12. Requisições

### 12.1. TRequisicoes

| Campo              | Tipo                    | Descrição | Obrigatório | Observação |
|--------------------|-------------------------|-----------|-------------|------------|
| Codigo             | String(6)               |           | SIM         |            |
| Estabelecimento    | String(4)               |           | SIM         |            |
| Requisitante       | String(6)               |           | SIM         |            |
| UsuarioResponsavel | String(10)              |           | NÃO         |            |
| CentroDeEstoque    | String(10)              |           | NÃO         |            |
| CentroDeResultados | String(10)              |           | NÃO         |            |
| Documento          | String(100)             |           | NÃO         |            |
| Data               | XSDatetime              |           | NÃO         |            |
| Hora               | XSDatetime              |           | NÃO         |            |
| Obs                | Blob                    |           | NÃO         |            |
| ItensRequisicao    | TArrayofItensRequisicao |           | NÃO         |            |
| Prioridade         |                         |           | NÃO         |            |

### 12.2. TItensRequisicao

| Campo                | Tipo       | Descrição | Obrigatório | Observação |
|----------------------|------------|-----------|-------------|------------|
| Codigo da Requisição | String(6)  |           | SIM         |            |
| Item                 | String(10) |           | SIM         |            |
| Quantidade           | Double     |           | SIM         |            |
| TotalProdutoComprado | Double     |           | NÃO         |            |



|                   |            |  |     |  |
|-------------------|------------|--|-----|--|
| Obs               | Blob       |  | NÃO |  |
| QuantidadeColeta  | Double     |  | NÃO |  |
| QuantidadePedida  | Double     |  | NÃO |  |
| QuantidadeEntrada | Double     |  | NÃO |  |
| ValorUltimaCompra | Double     |  | NÃO |  |
| DataEntrega       | XSDatetime |  | NÃO |  |

### 12.3. Métodos

- **IncluiRequisicoes**

- Assinatura:

- ```
function IncluiRequisicoes(const Requisicao: TRequisicoes): String; stdcall;
```

- Parâmetros:

- Objeto do tipo TRequisicoes;
 - Possíveis retornos:
 - Código da Requisição, caso a inclusão tenha sido bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **IncluiListaRequisicoes**

- Assinatura:

- ```
function IncluiListaRequisicoes(ListaRequisicao: TListaRequisicoes): TListaRequisicoes; stdcall;
```

- Parâmetros:

- Objeto do tipo TListaRequisicoes (é um array de objetos TRequisicoes);
  - Possíveis retornos:
    - Lista com o Código das Requisições incluídas;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **AlteraRequisicoes**

- Assinatura:

- ```
function AlteraRequisicoes(const Requisicao: TRequisicoes; const Codigo: String): String; stdcall;
```

- Parâmetros:

- Objeto do tipo TRequisicoes;
 - Código da Requisição;
 - Possíveis retornos:
 - Código da Requisição alterada;
 - Descrição do Erro, caso aconteça algum erro durante a alteração;
 - 'Requisição não cadastrada.' caso a requisição passada não seja encontrada;

- **AlteraListaRequisicoes**

- Assinatura:

- ```
function AlteraListaRequisicoes(ListaRequisicao: TListaRequisicoes): TListaRequisicoes; stdcall;
```

- Parâmetros:

- Objeto do tipo TListaRequisicoes (é um array de objetos TRequisicoes);
  - Possíveis retornos:
    - Lista de Requisições alteradas;

- Descrição do Erro, caso aconteça algum erro durante a alteração;
- **ExcluiRequisicoes**
    - Assinatura:  

```
function ExcluiRequisicoes(const Codigo: String): String;
stdcall;
```
    - Parâmetros:
      - Código da requisição;
    - Possíveis retornos:
      - 'OK', caso a requisição passada seja excluída com sucesso;
      - 'Requisição não encontrada.', caso a requisição passada não seja encontrada;
      - Descrição do Erro, caso aconteça algum erro durante a exclusão;
- **ExcluiListaRequisicoes**
    - Assinatura:  

```
function ExcluiListaRequisicoes(const ListaRequisicao:
TListaRequisicoes): String; stdcall;
```
    - Parâmetros:
      - Objeto do tipo TlistaRequisicoes (é um array de objetos TRequisicoes);
    - Possíveis retornos:
      - Lista de códigos das requisições excluídas;
      - Descrição do Erro, caso aconteça algum erro durante a exclusão;

## 13. Entradas

### 13.1. TEntradas

| Campo              | Tipo                 | Descrição | Obrigatório | Observação |
|--------------------|----------------------|-----------|-------------|------------|
| Estabelecimento    | String(4)            |           | SIM         |            |
| CentroDeEstoque    | String(10)           |           | SIM         |            |
| CentroDeResultados | String(10)           |           | NÃO         |            |
| Fornecedor         | String(6)            |           | SIM         |            |
| Pedido             | String(11)           |           | NÃO         |            |
| Transportadora     | String(6)            |           | NÃO         |            |
| Data               | XSDatetime           |           | NÃO         |            |
| Hora               | XSDatetime           |           | NÃO         |            |
| OBS                | Blob                 |           | NÃO         |            |
| Documento          | String(20)           |           | NÃO         |            |
| Valor              | Double               |           | NÃO         |            |
| Frete              | Double               |           | NÃO         |            |
| ItensEntrada       | TArrayOfItensEntrada |           | NÃO         |            |

### 13.2. TItensEntrada

| Campo      | Tipo       | Descrição | Obrigatório | Observação     |
|------------|------------|-----------|-------------|----------------|
| Código     | String(11) |           | SIM         |                |
| Sequencial | Integer    |           | SIM         | Autoincremento |
| Item       | String(10) |           | SIM         |                |
| Quantidade | Doube      |           | NÃO         |                |

|                          |           |                                                                           |     |                                            |
|--------------------------|-----------|---------------------------------------------------------------------------|-----|--------------------------------------------|
| LOT_Sequencial           | Integer   |                                                                           | NÃO |                                            |
| AliquotaIPI              | Double    |                                                                           | SIM | Caso não seja informado será incluído ZERO |
| AliquotaICMSSUBST        | Double    |                                                                           | SIM | Caso não seja informado será incluído ZERO |
| Frete                    | Double    |                                                                           | SIM | Caso não seja informado será incluído ZERO |
| ICMSSobreIPI             | Integer   | Indica se o ICMS incide sobre o IPI:<br>0 - NÃO<br>1 - SIM                | SIM | Caso não seja informado será incluído ZERO |
| AliqICMSDiferenciada     | Double    |                                                                           | NÃO |                                            |
| ICMSNaoDeduzTotal        | Integer   | Indica se o ICMS não deduz do total:<br>0 - NÃO<br>1 - SIM                | SIM | Caso não seja informado será incluído ZERO |
| ICMSBase                 | Double    |                                                                           | NÃO |                                            |
| AliquotaICMS             | Double    |                                                                           | NÃO |                                            |
| FreteCompoeBaseICMSSubst | Integer   | Indica se o frete compõe a base de ICMS substituto:<br>0 - NÃO<br>1 - SIM | SIM | Caso não seja informado será incluído ZERO |
| UnidadeEntrada           | String(3) |                                                                           | NÃO |                                            |
| FatorConversao           | Double    |                                                                           | NÃO |                                            |
| Acrescimo                | Double    |                                                                           | NÃO |                                            |
| Desconto                 | Double    |                                                                           | NÃO |                                            |
| Valor                    | Double    |                                                                           | NÃO |                                            |

### 13.3. Métodos

- **IncluiEntradas**

- Assinatura:

```
function IncluiEntradas(const Entrada: TEntradas): string;
stdcall;
```

- Parâmetros:

- Objeto do tipo TEntradas ;

- Possíveis retornos:

- Código da entrada incluída;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **AlteraEntradas**

- Assinatura:

```
function AlteraEntradas(const Entrada: TEntradas; const Codigo:
string): string; stdcall;
```

- Parâmetros:

- Objeto do tipo Tentradas;
    - Código da Entrada;

- Possíveis retornos:

- Código da entrada alterada;
    - 'Entrada não cadastrada', caso a entrada passada não seja encontrada;
    - Descrição do Erro, caso aconteça algum erro durante a alteração;

- **ExcluiEntradas**

- Assinatura:

```
function ExcluiEntradas(const Codigo: string): string;
stdcall;
```

- Parâmetros:

- Objeto do tipo Tentradas;
    - Código da Entrada;

- Possíveis retornos:

- Código da entrada alterada;
    - 'Entrada não cadastrada', caso a entrada passada não seja encontrada;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão;

## 14. Saídas

### 14.1. TSaidas

| Campo                    | Tipo               | Descrição | Obrigatório | Observação |
|--------------------------|--------------------|-----------|-------------|------------|
| Estabelecimento          | String(4)          |           | SIM         |            |
| Requisitante             | String(6)          |           | SIM         |            |
| CentroDeEstoque          | String(10)         |           | SIM         |            |
| CentroDeResultados       | String(10)         |           | SIM         |            |
| CentroDeResultadosOrigem | String(10)         |           | SIM         |            |
| Documento                | String(100)        |           | NÃO         |            |
| Data                     | XSDatetime         |           | NÃO         |            |
| Hora                     | XSDatetime         |           | NÃO         |            |
| Obs                      | Blob               |           | NÃO         |            |
| ItensSaida               | TArrayOfItensSaida |           | SIM         |            |

### 14.2. TItensSaida

| Campo          | Tipo       | Descrição | Obrigatório | Observação |
|----------------|------------|-----------|-------------|------------|
| Item           | String(10) |           | SIM         |            |
| Sequencial     | Integer    |           | SIM         |            |
| Quantidade     | Double     |           | SIM         |            |
| LOT_Sequencial | Integer    |           | SIM         |            |
| ContaAplicacao | String(30) |           | NÃO         |            |

### 14.3. Métodos

- **IncluiSaidas**

- Assinatura:

```
function IncluiSaidas(const Saida: TSaidas): String; stdcall;
```

- Parâmetros:

- Objeto do tipo TSaidas;

- Possíveis retornos:

- Código da saída incluída, caso a inclusão tenha sido realizada com sucesso;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **AlteraSaidas**

- Assinatura:

```
function AlteraSaidas(const Saida: TSaidas; const Codigo: String): String; stdcall;
```

- Parâmetros:

- Objeto do tipo TSaidas;
- Código da Saída;

- Possíveis retornos:

- Código da saída, caso a alteração tenha sido realizada com sucesso;
- Descrição do Erro, caso aconteça algum erro durante a alteração;

- **ExcluiSaidas**

- Assinatura:

```
function ExcluiSaidas(const Codigo: String): String; stdcall;
```

- Parâmetros:

- Código da Saída;

- Possíveis retornos:

- 'OK', caso a exclusão tenha sido realizada com sucesso;
- 'Saída não excluída', caso a saída não tenha sido encontrada;
- Descrição do Erro, caso aconteça algum erro durante a exclusão;

- **IncluirSaidasPadrao**

- Assinatura:

```
function IncluirSaidasPadrao(Data, Hora, CodigoEstabelecimento,
CodigoCentroDeEstoque, SaidaPadrao, CodigoCentroDeResultados,
CodigoCentroDeResultadosDestino, CodigoRequisitante,
QuantidadeDeSaidas: String): String; stdcall;
```

- Parâmetros:

- Data;
- Hora;
- Código do Estabelecimento;
- Código do Centro de Estoque;
- Saída Padrão;
- Código do Centro de Resultados;
- Código do Centro de Resultados Destino;
- Código do Requisitante;
- Quantidade de Saídas;

- Possíveis retornos:

- Código da saída, caso a alteração tenha sido realizada com sucesso;
- Descrição do Erro, caso aconteça algum erro durante a inclusão;

## 15. Devoluções de Saída

### 15.1. TDevolucoesSaida

| Campo                     | Tipo       | Descrição | Obrigatório | Observação |
|---------------------------|------------|-----------|-------------|------------|
| Estabelecimento           | String(4)  |           | SIM         |            |
| CentroDeEstoque           | String(10) |           | SIM         |            |
| CentroDeResultados        | String(10) |           | SIM         |            |
| CentroDeResultadosDestino | String(10) |           | SIM         |            |

|                      |                              |  |     |  |
|----------------------|------------------------------|--|-----|--|
| NumeroNotaFiscal     | String(47)                   |  | NÃO |  |
| Especie              | String(5)                    |  | NÃO |  |
| Data                 | XSDdateTime                  |  | NÃO |  |
| Hora                 | XSDdateTime                  |  | NÃO |  |
| OBS                  | Blob                         |  | NÃO |  |
| ItensDevolucoesSaida | TArrayOfItensDevolucoesSaida |  | SIM |  |

## 15.2. TItensDevolucoesSaida

| Campo          | Tipo       | Descrição | Obrigatório | Observação |
|----------------|------------|-----------|-------------|------------|
| Sequencial     | Integer    |           | SIM         |            |
| Item           | String(10) |           | SIM         |            |
| LOT_Sequencial | Integer    |           | SIM         |            |
| ContaAplicacao | String(30) |           | NÃO         |            |

## 15.3. Métodos

- **IncluiDevolucoesSaida**

- Assinatura:

```
function IncluiDevolucoesSaida(const DevolucoesSaida:
TDevolucoesSaida): string; stdcall;
```

- Parâmetros:

- Objeto do tipo TDevolucoesSaida;

- Possíveis retornos:

- Código da Devolução de Saída incluído, caso a inclusão tenha sido realizada com sucesso;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **AlteraDevolucoesSaida**

- Assinatura:

```
function AlteraDevolucoesSaida(const DevolucoesSaida:
TDevolucoesSaida; const Codigo: string): string; stdcall;
```

- Parâmetros:

- Objeto do tipo TDevolucoesSaida;
    - Código;

- Possíveis retornos:

- Código da Devolução de Saída, caso a alteração tenha sido realizada com sucesso;
    - 'Devolução de Saída não cadastrada.', caso a devolução passada não tenha sido encontrada;
    - Descrição do Erro, caso aconteça algum erro durante a alteração;

- **ExcluiDevolucoesSaida**

- Assinatura:

```
function ExcluiDevolucoesSaida(const Codigo: string): string; stdcall;
```

- Parâmetros:

- Código da Devolução de Saída;

- Possíveis retornos:

- 'OK', caso a exclusão tenha sido realizada com sucesso;
    - 'Devolução de Saída não excluída', caso a saída não tenha sido encontrada;
    - Descrição do Erro, caso aconteça algum erro durante a exclusão;

## 16. Pedido de Venda

### 16.1. TPedidoDeVenda

| Campo                 | Tipo                         | Descrição | Obrigatório | Observação                                          |
|-----------------------|------------------------------|-----------|-------------|-----------------------------------------------------|
| EST_CODIGO            | String(4)                    |           | SIM         |                                                     |
| CLI_Codigo            | String(6)                    |           | SIM         |                                                     |
| Data                  | XSDatetime                   |           | NÃO         |                                                     |
| DATAPREVENTREGA       | XSDatetime                   |           | NÃO         |                                                     |
| CONDICAOPAGAMENTO     | String(255)                  |           | NÃO         |                                                     |
| OBSERVACAO            | Blob                         |           | NÃO         |                                                     |
| Rejeitado             | Integer                      |           | SIM         |                                                     |
| TPP_Codigo            | String(4)                    |           | NÃO         | Chave da tabela de Prazo de Pagamento no Financeiro |
| TDC_Codigo            | String(3)                    |           | NÃO         | Chave da tabela de Tipo de Documento no Financeiro  |
| COB_Codigo            | String(4)                    |           | NÃO         | Chave da tabela de Agente Cobrador no Financeiro    |
| Documento             | String(100)                  |           | NÃO         |                                                     |
| BaselCMSSubst         | Double                       |           | NÃO         |                                                     |
| ValorICMSSubst        | Double                       |           | NÃO         |                                                     |
| USU_CODIGO            | String(10)                   |           | NÃO         |                                                     |
| HIS_CODIGO            | String(5)                    |           | NÃO         |                                                     |
| ServicosPedidoDeVenda | ArrayOfServicosPedidoDeVenda |           | (*)         |                                                     |
| ItensPedidoDeVenda    | ArrayOfItensPedidoDeVenda    |           | (*)         |                                                     |

(\*) obrigatório ter ServicosPedidoDeVenda OU ItensPedidoDeVenda

### 16.2. TItensPedidoDeVenda

| Campo         | Tipo       | Descrição   | Obrigatório | Observação                                 |
|---------------|------------|-------------|-------------|--------------------------------------------|
| PEV_CODIGO    | String(11) |             | SIM         |                                            |
| PRO_CODIGO    | String(10) |             | SIM         |                                            |
| SEQUENCIAL    | Integer    |             | SIM         |                                            |
| QTDE          | Double     |             | SIM         |                                            |
| VALOR         | Double     |             | SIM         |                                            |
| AliquotaICMS  | Double     |             | NÃO         |                                            |
| AliquotaIPI   | Double     |             | NÃO         |                                            |
| CFOP_Codigo   | String(4)  |             | NÃO         | Chave da tabela de CFOP (é o próprio CFOP) |
| STA_Codigo    | String(1)  |             | NÃO         | Chave da tabela de Situação Tributária A   |
| STB_Codigo    | String(2)  |             | NÃO         | Chave da tabela de Situação Tributária B   |
| Unidade       | String(3)  |             | NÃO         |                                            |
| Frete         | Double     |             | NÃO         |                                            |
| COMPDESCRICAO | Blob       | Complemento | NÃO         |                                            |

|                |           | da Descrição |     |                                                                  |
|----------------|-----------|--------------|-----|------------------------------------------------------------------|
| Comprimento    | Double    |              | NÃO |                                                                  |
| Altura         | Double    |              | NÃO |                                                                  |
| Largura        | Double    |              | NÃO |                                                                  |
| Desconto       | Double    |              | NÃO |                                                                  |
| STBSN_Codigo   | String(3) |              | NÃO | Chave da tabela de Situação Tributária B para o Simples Nacional |
| AliqICMSSubst  | Double    |              | NÃO |                                                                  |
| BaseICMSSubst  | Double    |              | NÃO |                                                                  |
| ValorICMSSubst | Double    |              | NÃO |                                                                  |

### 16.3. TServicePedidoDeVenda

| Campo       | Tipo       | Descrição | Obrigatório | Observação |
|-------------|------------|-----------|-------------|------------|
| PEV_CODIGO  | String(11) |           | SIM         |            |
| PRO_CODIGO  | String(10) |           | SIM         |            |
| Sequencial  | Integer    |           | SIM         |            |
| SER_CODIGO  | String(4)  |           | SIM         |            |
| MDS_Codigo  | String(2)  |           | SIM         |            |
| Valor       | Double     |           | SIM         |            |
| Observacao  | Blob       |           | NÃO         |            |
| QTDSERVICOS | Double     |           | NÃO         |            |
| DESCONTO    | Double     |           | NÃO         |            |

### 16.4. Métodos

- **IncluiPedidoDeVenda**
  - Assinatura:  

```
function IncluiPedidoDeVenda(const PedidoDeVenda: TPedidoDeVenda): String; stdcall;
```
  - Parâmetros:
    - Objeto do tipo TPedidoDeVenda;
  - Possíveis retornos:
    - Código do Pedido de Venda incluído, caso a inclusão tenha sido realizada com sucesso;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão;
- **AlteraPedidoDeVenda**
  - Assinatura:  

```
function AlteraPedidoDeVenda(const PedidoDeVenda: TPedidoDeVenda; const Codigo: String): String; stdcall;
```
  - Parâmetros:
    - Objeto do tipo TPedidoDeVenda;
    - Código do Pedido de Venda;
  - Possíveis retornos:
    - Código do Pedido de Venda, caso a alteração tenha sido realizada com sucesso;
    - 'Pedido de Venda não cadastrado', caso o pedido de venda passado não seja encontrado;
    - Descrição do Erro, caso aconteça algum erro durante a alteração;
- **ExcluiPedidoDeVenda**



- Assinatura:  

```
function ExcluiPedidoDeVenda(const Codigo: String): String;
stdcall;
```
- Parâmetros:
  - Código do Pedido de Venda;
- Possíveis retornos:
  - 'OK', caso a exclusão tenha sido realizada com sucesso;
  - 'Pedido de Venda não cadastrado', caso o pedido de venda não tenha sido encontrado;
  - Descrição do Erro, caso aconteça algum erro durante a exclusão;

- **ReplicaPedidoDeVenda**

- Assinatura:  

```
function ReplicaPedidoDeVenda(const Codigo: String; QtdPeriodo,
Periodicidade: Integer; TipoPeriodo: String; BaseICMSSubst: Double =
0; ValorICMSSubst: Double = 0): String; stdcall;
```
- Parâmetros:
  - Código;
  - QtdPeriodo;
  - Periodicidade;
  - TipoPeriodo;
  - BaseICMSSubst;
  - ValorICMSSubst;
- Possíveis retornos:
  - 'Pedido de Venda replicado com sucesso.', caso a cópia tenha sido realizada com sucesso;
  - 'Pedido de Venda não replicado.', caso o pedido de venda passado não tenha sido encontrado;
  - Descrição do Erro, caso aconteça algum erro durante a cópia do pedido;

## 17. Receitas e Despesas

### 17.1. TReceitasDespesas

| Campo         | Tipo       | Descrição                                                                | Obrigatório | Observação                               |
|---------------|------------|--------------------------------------------------------------------------|-------------|------------------------------------------|
| Codigo        | String(10) |                                                                          | SIM         |                                          |
| Nome          | String(50) |                                                                          | SIM         |                                          |
| Tipo          | String(1)  | 'A' para Analítico<br>'S' para Sintético                                 | SIM         |                                          |
| Natureza      | String(1)  | 'E' para Entrada<br>'S' para saída                                       | NÃO         |                                          |
| Inativo       | Integer    | Indica se a conta de receita/despesa está inativa:<br>0 - NÃO<br>1 - SIM | SIM         | Se não informar é incluído '0-NÃO'       |
| CodigoMae     | String(10) |                                                                          | NÃO         |                                          |
| ContaContabil | String(15) |                                                                          | NÃO         |                                          |
| Grupo         | String(4)  |                                                                          | NÃO         | Chave da tabela Grupo de Receita/Despesa |
| Investimento  | Integer    | Indica se a                                                              | SIM         | Se não informar é                        |

|                      |            |                                                                         |     |                                    |
|----------------------|------------|-------------------------------------------------------------------------|-----|------------------------------------|
|                      |            | despesa é vinculada a apropriação (Investimento):<br>0 - NÃO<br>1 - SIM |     | incluído '0-NÃO'                   |
| RetemISS             | Integer    | Indica se retem ISS:<br>0 - NÃO<br>1 - SIM                              | SIM | Se não informar é incluído '0-NÃO' |
| RetemINSS            | Integer    | Indica se retem INSS:<br>0 - NÃO<br>1 - SIM                             | SIM | Se não informar é incluído '0-NÃO' |
| RetemIRRF            | Integer    | Indica se retem IRRF:<br>0 - NÃO<br>1 - SIM                             | SIM | Se não informar é incluído '0-NÃO' |
| RetemPISCOFINS/CSLL  | Integer    | Indica se retem PIS/COFINS/CSLL:<br>0 - NÃO<br>1 - SIM                  | SIM | Se não informar é incluído '0-NÃO' |
| ContaFinanceira      | String(4)  |                                                                         | NÃO |                                    |
| ContaContabilEstoque | String(15) |                                                                         | NÃO |                                    |
| GeraSPR              | Integer    | Indica se Gera Serviços Prestados:<br>0 - NÃO<br>1 - SIM                | SIM | Se não informar é incluído '0-NÃO' |
| RetemICMS            | Integer    | Indica se retem PIS/COFINS/CSLL:<br>0 - NÃO<br>1 - SIM                  | SIM | Se não informar é incluído '0-NÃO' |
| Obs                  | Blob       |                                                                         | NÃO |                                    |

## 17.2. Métodos

- **IncluiReceitaDespesa**

- Assinatura:

```
function IncluiReceitaDespesa(const ReceitaDespesa:
TReceitasDespesas): String; stdcall;
```

- Parâmetros:

- Objeto do tipo TReceitasDespesas;

- Possíveis retornos:

- Código da Receita/Despesa, caso a inclusão seja bem sucedida;
    - 'Não foi possível incluir Conta de Receita/Despesa. (Não foi passado um tipo válido para a função).', caso não seja possível carregar a Receita/Despesa passada;
    - 'Não conseguiu acessar o parâmetro repassado para a função: ReceitaDespesa.Codigo.', caso não consiga acessar a Receita/Despesa passada;
    - 'Não encontrou Conta de Receita/Despesa', caso a Receita/Despesa passada não seja encontrada;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **AlteraReceitaDespesa**

- Assinatura:

```
function AlteraReceitaDespesa(const ReceitaDespesa:
TReceitasDespesas): String; stdcall;
```

- Parâmetros:
    - Objeto do tipo TReceitasDespesas;
  - Possíveis retornos:
    - Código da Receita/Despesa, caso a alteração seja bem sucedida;
    - 'Não foi possível incluir Conta de Receita/Despesa. (Não foi passado um tipo válido para a função).', caso não seja possível carregar a Receita/Despesa passada;
    - 'Não conseguiu acessar o parâmetro repassado para a função: ReceitaDespesa.Codigo.', caso não consiga acessar a Receita/Despesa passada;
    - 'Conta de Receita/Despesa não cadastrada.', caso a Receita/Despesa passada não seja encontrada;
    - Descrição do Erro, caso aconteça algum erro durante a alteração;
- **ExcluiReceitaDespesa**
    - Assinatura:
 

```
function ExcluiReceitaDespesa(const ReceitaDespesa:
TReceitasDespesas): String; stdcall;
```
    - Parâmetros:
      - Objeto do tipo TReceitasDespesas;
    - Possíveis retornos:
      - 'Conta de Receita/Despesa Excluída', caso a exclusão seja bem sucedida;
      - 'Não foi possível incluir Conta de ReceitaDespesa. (Não foi passado um tipo válido para a função).', caso não seja possível carregar a ReceitaDespesa passada;
      - 'Não conseguiu acessar o parâmetro repassado para a função: ReceitaDespesa.Codigo.', caso não consiga acessar a Receita/Despesa passada;
      - 'Conta de Receita/Despesa não cadastrada.', caso a Receita/Despesa passada não seja encontrada;
      - Descrição do Erro, caso aconteça algum erro durante a exclusão;
  - **GetObjetoReceitasDespesas**
    - Assinatura:
 

```
function GetObjetoReceitasDespesas(const Codigo: string):
TReceitasDespesas; stdcall;
```
    - Parâmetros:
      - Código da Receita/Despesa;
    - Possíveis retornos:
      - *nil*, caso a Receita/Despesa não seja passada;
      - Objeto do tipo TreceitasDespesas;
      - Descrição do Erro, caso aconteça algum erro durante a pesquisa;

## 18. Centro de Resultados

### 18.1. TCentroDeResultado

| Campo          | Tipo       | Descrição | Obrigatório | Observação |
|----------------|------------|-----------|-------------|------------|
| Codigo         | String(10) |           | SIM         |            |
| Nome           | String(40) |           | SIM         |            |
| CRS_Codigo_Mae | String(10) |           | NÃO         |            |

|               |            |  |     |                                               |
|---------------|------------|--|-----|-----------------------------------------------|
| ContaContabil | String(15) |  | NÃO |                                               |
| GCR_Codigo    | String(4)  |  | NÃO | Chave da tabela Grupo de Centro de Resultados |
| Inativo       | Integer    |  | SIM | Se não informar é incluído '0-NÃO'            |

## 18.2. Métodos

- **IncluiCentroDeResultado**

- Assinatura:

```
function IncluiCentroDeResultado(CentroDeResultado:
TCentroDeResultado): string; stdcall;
```

- Parâmetros:

- Objeto do tipo TCentroDeResultado;

- Possíveis retornos:

- Código do Centro de Resultado, caso a inclusão seja bem sucedida;
    - 'Não foi possível incluir Centro de Resultado. (Não foi passado um tipo válido para a função).', caso não seja possível carregar o Centro de Resultado passado;
    - 'Não conseguiu acessar o parâmetro repassado para a função: CentroDeResultado.Codigo.', caso não consiga acessar o Centro de Resultado passado;
    - 'Não encontrou Centro de Resultado', caso o Centro de Resultado passado não seja encontrado;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **IncluiArrayCentroDeResultado**

- Assinatura:

```
function IncluiArrayCentroDeResultado (ArrayCentroDeResultado:
TArrayCentroDeResultado): TArrayCentroDeResultado; stdcall;
```

- Parâmetros:

- Objeto do tipo TArrayCentroDeResultado;

- Possíveis retornos:

- Array dos Centro de Resultados incluídos, caso a inclusão seja bem sucedida;
    - 'Não foi possível incluir a Lista de Centro de Resultado. (Não foi passado um tipo válido para a função).', caso o objeto passado seja *nil*;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão;

- **AlteraCentroDeResultado**

- Assinatura:

```
function AlteraCentroDeResultado(CentroDeResultado:
TCentroDeResultado): string; stdcall;
```

- Parâmetros:

- Objeto do tipo TCentroDeResultado;

- Possíveis retornos:

- Código do Centro de Resultado, caso a inclusão seja bem sucedida;
    - 'Não foi possível alterar Centro de Resultado. (Não foi passado um tipo válido para a função).', caso não seja possível carregar o Centro de Resultado passado;
    - 'Não conseguiu acessar o parâmetro repassado para a função: CentroDeResultado.Codigo.', caso não consiga acessar o Centro de Resultado passado;

- 'Centro de Resultado não cadastrado', caso o Centro de Resultado passado não seja encontrado;
  - Descrição do Erro, caso aconteça algum erro durante a alteração;
- **AlteraArrayCentroDeResultado**
  - Assinatura:
 

```
function AlteraArrayCentroDeResultado (ArrayCentroDeResultado: TArrayCentroDeResultado): TArrayCentroDeResultado; stdcall;
```
  - Parâmetros:
    - Objeto do tipo TArrayCentroDeResultado;
  - Possíveis retornos:
    - Array dos Centro de Resultados alterados, caso a alteração seja bem sucedida;
    - 'Não foi possível alterar a Lista de Centro de Resultado. (Não foi passado um tipo válido para a função).', caso o objeto passado seja *nil*;
    - Descrição do Erro, caso aconteça algum erro durante a alteração;
- **ExcluiCentroDeResultado**
  - Assinatura:
 

```
function ExcluiCentroDeResultado(CentroDeResultado: TCentroDeResultado): string; stdcall;
```
  - Parâmetros:
    - Objeto do tipo TCentroDeResultado;
  - Possíveis retornos:
    - 'Centro de Resultado excluído', caso a exclusão seja bem sucedida;
    - 'Não foi possível excluir Centro de Resultado. (Não foi passado um tipo válido para a função).', caso não seja possível carregar o Centro de Resultado passado;
    - 'Não conseguiu acessar o parâmetro repassado para a função: CentroDeResultado.Codigo.', caso não consiga acessar o Centro de Resultado passado;
    - 'Centro de Resultado não cadastrado', caso o Centro de Resultado passado não seja encontrado;
    - Descrição do Erro, caso aconteça algum erro durante a exclusão;
- **ExcluiArrayCentroDeResultado**
  - Assinatura:
 

```
function ExcluiArrayCentroDeResultado (ArrayCentroDeResultado: TArrayCentroDeResultado): Boolean; stdcall;
```
  - Parâmetros:
    - Objeto do tipo TArrayCentroDeResultado;
  - Possíveis retornos:
    - **True**, caso a exclusão seja bem sucedida;
    - 'Não foi possível excluir a Lista de Centro de Resultado. (Não foi passado um tipo válido para a função).', caso o objeto passado seja *nil*;
    - Descrição do Erro, caso aconteça algum erro durante a alteração;
- **GetObjetoCentroDeResultado**
  - Assinatura:
 

```
function GetObjetoCentroDeResultado(const Codigo: string): TCentroDeResultado; stdcall;
```
  - Parâmetros:

- Código do Centro de Resultado;
- Possíveis retornos:
  - *nil*, caso o Centro de Resultados não seja passado;
  - Objeto do tipo TCentroDeResultado;
  - Descrição do Erro, caso aconteça algum erro durante a pesquisa;

## Interfaces disponíveis no *WebService* – Integração JSON

Todas as estruturas das classes citadas nesta seção podem ser encontradas na seção específica de cada classe.

**IMPORTANTE:** A estrutura do Json é *Case Sensitive*.

### 1. Cliente

#### 1.1. Estrutura do JSON

```
{
 "codigo": "string",
 "nome": "string",
 "nomeFantasia": "string",
 "cnpjcpf": "string",
 "cep": "string",
 "logradouro": "string",
 "bairro": "string",
 "numero": "string",
 "complemento": "string",
 "uf": "string",
 "cidade": "string",
 "ddd": "string",
 "fone": "string",
 "dddfax": "string",
 "fax": "string",
 "identificador": "string",
 "dddcelular": "string",
 "celular": "string",
 "email": "string",
 "contato": "string",
 "site": "string",
 "rg": "string",
 "ie": "string",
 "grupo": "string",
 "agenteCobrador": "string",
 "receita": "string",
 "contaContabil": "string",
 "Representante": "string",
 "Vencimento": "integer",
 "Etiqueta": "integer",
 "GeraNFAuto": "integer",
 "RetemISS": "integer",
 "RetemINSS": "integer",
 "AliquotaISS": "integer",
 "CentroResultado": "string",
 "IM": "string",
 "CampoExtra1": "string",
 "CampoExtra2": "string",
}
```

```

 "CampoExtra3": "string",
 "CampoExtra4": "string",
 "CampoExtra5": "string",
 "CampoExtra6": "string",
 "CampoExtra7": "string",
 "CampoExtra8": "string",
 "CampoExtra9": "string",
 "CampoExtra10": "string",
 "IDWS": "string",
 "CLIPagador": "string",
 "EhCLIPagador": "integer",
 "Exterior": "integer",
 "IsentoIE": "integer",
 "IndicadorIEDestinatarior": "integer",
 "DocumentoEstrangeiro": "string",
 "GrupoEconomico": "string"
}

```

## 1.2. Métodos

- **IncluirClienteComJSON**

- Assinatura:
 

```
function IncluirClienteComJSON(const JSON: String): String;
stdcall;
```
- Parâmetros:
  - Json do objeto TCliente;
- Possíveis retornos:
  - string no formato JSON;

- **IncluirListaClienteComJSON**

- Assinatura:
 

```
function IncluirListaClienteComJSON(const ListaJSON:
String):String; stdcall;
```
- Parâmetros:
  - Json do objeto TAgendamentoCliente (é um array de objetos TCliente);
- Possíveis retornos:
  - string no formato JSON;

- **AlterarClienteComJSON**

- Assinatura:
 

```
function AlterarClienteComJSON(const JSON: String): String;
stdcall;
```
- Parâmetros:
  - Json do objeto TCliente;
- Possíveis retornos:
  - string no formato JSON;

- **AlterarListaClienteComJSON**

- Assinatura:
 

```
function AlterarListaClienteComJSON(const ListaJSON: String):
String; stdcall;
```
- Parâmetros:

- Json do objeto TagrupamentoCliente (é um array de objetos TCliente);
- Possíveis retornos:
  - string no formato JSON;
- **ExcluirClienteComJSON**
  - Assinatura:
 

```
function ExcluirClienteComJSON(const JSON: string): String;
stdcall;
```
  - Parâmetros:
    - Json do objeto TCliente;
  - Possíveis retornos:
    - 'OK', caso a exclusão seja bem sucedida;
    - Descrição do Erro, caso aconteça algum erro durante a exclusão;
- **GetJsonClientePorCodigo**
  - Assinatura:
 

```
function GetJsonClientePorCodigo(const Codigo: String):
String;stdcall;
```
  - Parâmetros:
    - Código do Cliente;
  - Possíveis retornos:
    - string do Json do objeto TCliente, caso a busca tenha sido bem sucedida;
    - mensagem informando que o cliente não foi encontrado;

## 2. Fornecedor

### 2.1. Estrutura do JSON

```
{
 "Codigo": "string",
 "Nome": "string",
 "NomeFantasia": "string",
 "CNPJCPF": "string",
 "CEP": "string",
 "Logradouro": "string",
 "Bairro": "string",
 "Numero": "string",
 "Complemento": "string",
 "UF": "string",
 "Cidade": "string",
 "DDD": "string",
 "Fone": "string",
 "Fax": "string",
 "Email": "string",
 "Contato": "string",
 "Site": "string",
 "RG": "string",
 "IE": "string",
 "IndicadorIE": "integer",
 "Grupo": "string",
 "Receita": "string",
 "IM": "string",
 "Referencia": "",
```



```

 "FinalidadeTED": "string",
 "FinalidadeDOC": "string",
 "Celular": "string",
 "Leiaute": "string",
 "Banco": "string",
 "BancoNumero": "string",
 "Agencia": "string",
 "CC": "string",
 "DVAg": "string",
 "DVCC": "string",
 "ServicoPublico": "integer",
 "ICMS": "integer",
 "IPI": "integer",
 "IDWS": "string"
}

```

## 2.2. Métodos

- **IncluirFornecedorComJSON**

- Assinatura:

```
function IncluirFornecedorComJSON(const JSON: String):
String; stdcall;
```

- Parâmetros:

- Json do objeto TForneecedor;

- Possíveis retornos:

- string no formato JSON;

- **IncluirListaFornecedorComJSON**

- Assinatura:

```
function IncluirListaFornecedorComJSON(const ListaJSON:
String):String; stdcall;
```

- Parâmetros:

- Json do objeto TAgendamentoFornecedor (é um array de objetos TForneecedor);

- Possíveis retornos:

- string no formato JSON;

- **AlterarFornecedorComJSON**

- Assinatura:

```
function AlterarFornecedorComJSON(const JSON: String): String;
stdcall;
```

- Parâmetros:

- Json do objeto TForneecedor;

- Possíveis retornos:

- string no formato JSON;

- **AlterarListaFornecedorComJSON**

- Assinatura:

```
function AlterarListaFornecedorComJSON(const ListaJSON: String):
String; stdcall;
```

- Parâmetros:

- Json do objeto TAgendamentoFornecedor (é um array de objetos TForneecedor);

- Possíveis retornos:
  - string no formato JSON;
- **ExcluirFornecedorComJSON**
  - Assinatura:
 

```
function ExcluirFornecedorComJSON(const JSON: String): String;
stdcall;
```
  - Parâmetros:
    - Json do objeto TProveedor;
  - Possíveis retornos:
    - 'OK', caso a exclusão seja bem sucedida;
    - Descrição do Erro, caso aconteça algum erro durante a exclusão;

### 3. Serviço [Não implementado]

Entre em contato com o desenvolvimento da Fortes Tecnologia caso precise desta implementação.

## 4. Item

### 4.1. Estrutura do JSON

```
{
 "Valor": "double",
 "Codigo": "string",
 "Nome": "string",
 "Unidade": "string",
 "ICMSSubstTipo": "string",
 "CompDescricao": "string",
 "CodUtilizadoESTFiscal": "string",
 "CodigoNCM": "string",
 "CodigoBarra": "string",
 "CodigoBarraUnidadeTribut": "string",
 "CodigoNVE": "string",
 "CodigoANP": "string",
 "CodProFortesFiscal": "integer",
 "RegistroAnvisa": "string",
 "Despesa": "string",
 "Referencia": "string",
 "UtilizarAliquotaNCM": "integer",
 "ClassificacaoFiscal": "string",
 "DescricaoUnidade": "string",
 "CalculaICMS": "integer",
 "DeducaoICMS": "integer",
 "CalculaIRRF": "integer",
 "CalculaPIS": "integer",
 "CalculaCOFINS": "integer",
 "CalculaCSL": "integer",
 "AliquotaIPI": "integer",
 "SitTribIPISaída": "string",
 "SitTribCOFINSSaída": "string",
 "SitTribPISSaída": "string",
 "NaturezaReceitaCOFINS": "integer",
 "NaturezaReceitaPIS": "integer",
}
```

```

 "ICMSSubstituicao": "integer",
 "AliquotaICMSSubstAgregacao": "integer",
 "ValorPautaICMSSubst": "integer",
 "IPIFixo": "integer",
 "ValorIPIFixo": "integer",
 "Categoria": "string",
 "Acabamento": "string",
 "Material": "string"
}

```

## 4.2. Métodos

- **IncluirItemComJSON**

- Assinatura:

```

function IncluirItemComJSON(const JSON: String): String;
stdcall;

```

- Parâmetros:

- Json do objeto TItem;

- Possíveis retornos:

- string no formato JSON;

- **IncluirListaItemComJSON**

- Assinatura:

```

function IncluirListaItemComJSON(const ListaJSON: String):
String; stdcall;

```

- Parâmetros:

- Json do objeto TAgupamentoItem (é um array de objetos TItem);

- Possíveis retornos:

- string no formato JSON;

- **AlterarItemComJSON**

- Assinatura:

```

function AlterarItemComJSON(const JSON: String): String;
stdcall;

```

- Parâmetros:

- Json do objeto TItem;

- Possíveis retornos:

- string no formato JSON;

- **ExcluirItemComJSON**

- Assinatura:

```

function ExcluirItemComJSON(const JSON: String): String;
stdcall;

```

- Parâmetros:

- Json do objeto TItem;

- Possíveis retornos:

- 'OK', caso a exclusão seja bem sucedida;

- Descrição do Erro, caso aconteça algum erro durante a exclusão;

- **GetListaItensComPaginacaoJSON**

- Assinatura:

```
function GetListaItensComPaginacaoJSON(CES_Codigo: String = '';
GIT_Codigo: String = ''; QuantidadeRegistronaPagina: Integer = 0;
NumerodaPagina: Integer = 0):String;stdcall;
```

- Parâmetros(Opcionais):

CES\_Codigo (Código do centro de estoque), GIT\_Codigo(Código do grupo de Itens),  
QuantidadeRegistronaPagina<sup>1</sup> (Quantidade de registro que deseja buscar) e  
NumerodaPagina<sup>2</sup> (Número da página que deseja buscar).

- Possíveis retornos:

String no formato JSON com listagem de itens e com um atributo("PROXIMA\_PAGINA")  
indicando a próxima página que pode ser consultada com os parâmetros de paginação  
informados.<sup>3</sup>

String no formato JSON com "Nenhum item encontrado para os parâmetros solicitados",  
indicando que não itens com os parâmetros informados.

<sup>1</sup> Se esse parâmetro não passado ou for igual a zero, a consulta será feita sem paginação.

<sup>2</sup> Se esse parâmetro não passado ou for igual a zero, a consulta será feita sem paginação.

<sup>3</sup> Caso o atributo "PROXIMA\_PAGINA" seja igual a 0 (zero), indica que não há mais páginas com dados a  
serem consultadas, esse atributo "PROXIMA\_PAGINA" só será retornado se os parâmetros  
"QuantidadeRegistronaPagina" e "NumerodaPagina" forem maiores que zero(0), do contrário o json  
retorna apenas terá o atributo "ITENS" com listagem de itens.

Exemplo de retorno:

```
{
 "PROXIMA_PAGINA": "2",
 "ITENS": [{
 "CODIGO": "0000000001",
 "NOME": "PRODUTO DE TESTE",
 "VALOR": "30",
 "UNIDADE": "UN",
 "ICMSSUBSTTIPO": "A",
 "COMPDESCRICAO": "TESTE CMP450L 01729585 KF",
 "CODUTILIZADOESTFISCAL": "0000000001",
 "CODIGONCM": "84271011",
 "GIT_CODIGO": "0002",
 "CODPROFISCAL": "1234",
 "CRD_CODIGO": "3020101003",
 "REFERENCIA": "",
 "CLASSIFICACAOFISCAL": "84271019",
 "DESCRICAOUNIDADE": "UNIDADE",
 "ICMS": "1",
 "DEDUCAOICMS": "0",
 "IRRF": "0",
 "PIS": "1",
 "COFINS": "1",
 "CSL": "1",
 "ALIQUOTAIP": "0",
 "STI_CODIGO": "",
 "STF_CODIGO": "08",
 "STS_CODIGO": "08",
 "ICMSSUBSTITUICAO": "0",
```

```

 "ICMSSUBSTAGREGACAO": "8",
 "ICMSSUBSTVALORPAUTA": "0",
 "MIT_CODIGO": "",
 "CFOP_CODIGODENTROESTADO": "5949",
 "STA_CODIGO": "0",
 "STB_CODIGO": "41",
 "NCM_CODIGO": "008056",
 "UNIDADEENTRADA": "UN",
 "TIT_CODIGO": "",
 "IPI": "0",
 "IDWS": "",
 "STBSN_CODIGO": "",
 "STBSN_CODIGOFORAESTADO": "",
 "STBSN_CODIGOFORAPAIS": "",
 "STBSN_CODIGO_NFPROPRIA": "",
 "STBSN_FORAESTADO_NFPROPRIA": "",
 "STBSN_FORAPAIS_NFPROPRIA": "",
 "STBSN_CODIGO_NFE": "",
 "STBSN_CODIGOFORAESTADO_NFE": "",
 "STBSN_CODIGOFORAPAIS_NFE": "",
 "CUSTO_MEDIO": 54.65
 }
}

```

## 5. Lançamento

### 5.1. Estrutura do JSON

```

{
 "Codigo": "string",
 "Estabelecimento": "string",
 "Conta": "string",
 "Data": "string",
 "Sequencial": "integer",
 "CentroResultados": "string",
 "ReceitaDespesa": "string",
 "Valor": "double",
 "Natureza": "string",
 "ExportaAC": "integer",
 "Historico": "string",
 "RateioLancamento": [
 {
 "Codigo": "string",
 "Sequencial": integer,
 "Estabelecimento": "string",
 "CentroResultados": "string",
 "ReceitaDespesa": "string",
 "Natureza": "string",
 "Valor": "integer",
 "Historico": "string"
 }
]
}

```

## 5.2. Métodos

- **IncluirLancamentoComJSON**

- Assinatura:

- ```
function IncluirLancamentoComJSON(const JSON: string): string;
stdcall;
```

- Parâmetros:

- Json do objeto TLançamento;

- Possíveis retornos:

- string no formato JSON;

- **IncluirListaLancamentoComJSON**

- Assinatura:

- ```
function IncluirListaLancamentoComJSON(const ListaJSON: string):
string; stdcall;
```

- Parâmetros:

- Json do objeto TAgendamentoLancamento (é um array de objetos do tipo TLancamento);

- Possíveis retornos:

- string no formato JSON;
    - Descrição do Erro, caso aconteça algum erro durante a inclusão da lista;

- **ExcluirLancamentoComJSON**

- Assinatura:

- ```
function ExcluirLancamentoComJSON(const JSON: string): string;
stdcall;
```

- Parâmetros:

- Json do objeto TLançamento;

- Possíveis retornos:

- string no formato JSON;
 - Descrição do Erro, caso aconteça algum erro durante a exclusão;

- **ExcluirListaLancamentoComJSON**

- Assinatura:

- ```
function ExcluirListaLancamentoComJSON(const JSON: string):
string; stdcall;
```

- Parâmetros:

- Json do objeto TAgendamentoLancamento (é um array de objetos do tipo TLancamento);

- Possíveis retornos:

- string no formato JSON;
    - Descrição do Erro, caso aconteça algum erro durante a exclusão;

- **GetJsonLancamentoPorCodigo**

- Assinatura:

- ```
function GetJsonLancamentoPorCodigo(const Codigo: String):
String; stdcall;
```

- Parâmetros:

- Código do Lançamento;

- Possíveis retornos:
 - string no formato JSON, caso a busca seja bem sucedida;
 - Descrição do Erro, caso aconteça algum erro durante a busca;

6. Contas a Receber

6.1. Estrutura do JSON

6.1.1. Contas a Receber

```
{
  "codigo": "20220100103",
  "estabelecimento": "0001",
  "centroResultados": "",
  "cliente": "03744126000169",
  "codigocliente": "000001",
  "receita": "",
  "documento": "",
  "tipoGeracao": "M",
  "emissao": "2022-01-01",
  "exportaAC": 1,
  "ISS": 75.01,
  "IRRF": 0,
  "INSS": 0,
  "PIS": 0,
  "DescontoIncondicional": 0,
  "COFINS": 0,
  "CSLL": 0,
  "obs": "",
  "mesAno": "",
  "vencimentosaReceber": [
    {
      "codigo": "20220100103",
      "sequencial": 1,
      "vencimento": "2022-01-01",
      "valor": 674.99,
      "valorOriginal": 674.99,
      "tipoDocumento": "001",
      "titulo": "",
      "agenteCobrador": "0007",
      "nrBoleto": "",
      "idBloquete": 0,
      "descontoPrevisto": 0,
      "dtCancelamento": "1899-12-30",
      "Quitado": 0,
      "DataVencimento": "01/01/2022",
      "DataCancelamento": "",
      "Obs": "",
      "FormaPagamento": "",
      "CodigoBarras": "",
      "LinhaDigitada": "",
      "PercentualJuros": 0,
      "PercentualMulta": 0
    },
    {
      "codigo": "20220100103",
```

```

        "sequencial": 2,
        "vencimento": "2022-02-01",
        "valor": 750.2,
        "valorOriginal": 750.2,
        "tipoDocumento": "001",
        "titulo": "",
        "agenteCobrador": "0001",
        "nrBoleto": "",
        "idBloquete": 0,
        "descontoPrevisto": 0,
        "dtCancelamento": "1899-12-30",
        "Quitado": 0,
        "DataVencimento": "01/02/2022",
        "DataCancelamento": "",
        "Obs": "",
        "FormaPagamento": "",
        "CodigoBarras": "",
        "LinhaDigitada": "",
        "PercentualJuros": 0,
        "PercentualMulta": 0
    }
],
"vencimentosChequeaReceber": [],
"vencimentosCartaoaReceber": [],
"servicosaReceber": [
    {
        "codigo": "20220100103",
        "sequencial": 1,
        "servico": "0001",
        "modalidade": "01",
        "obs": "observacao do servico",
        "valor": 1500.2,
        "comissionadosservicosareceber": [],
        "DescontoIncondicional": 0,
        "ValorBruto": 0,
        "AliquotaISS": 0,
        "Quantidade": 0
    }
],
"itensaReceber": [],
"rateiosaReceber": [
    {
        "Codigo_CRE": "20220100103",
        "Estabelecimento": "0001",
        "CentroDeResultados": "001",
        "Receita": "4010010003",
        "Observacao": "",
        "Valor": 750
    },
    {
        "Codigo_CRE": "20220100103",
        "Estabelecimento": "0001",
        "CentroDeResultados": "002",
        "Receita": "4010010003",
        "Observacao": "",
    }
]

```



```

        "Valor": 750.2
    }
],
"IDWS": "",
"ClienteIDWS": "",
"CampoExtra1": "",
"CampoExtra2": "",
"CampoExtra3": "",
"CampoExtra4": "",
"CampoExtra5": "",
"CampoExtra6": "",
"CampoExtra7": "",
"CampoExtra8": "",
"CampoExtra9": "",
"CampoExtra10": "",
"ContaAplicacao": "",
"DataEmissao": "01/01/2022",
"GreCodigo": "",
"OutrasRetencoes": 0,
"HistoricoPadrao": "",
"ContaContabilCliente": ""
}

```

6.1.2. Baixa do Contas a Receber

```

{
  "ArrayBaixaContasaReceber": [
    {
      "CRE_Codigo": "String",
      "SequencialVencimento": "Integer",
      "Sequencial": "Integer",
      "ExportaAC": "Integer",
      "Data": "TDateTime",
      "Valor": "Double",
      "Desconto": "Double",
      "Juros": "Double",
      "Multa": "Double",
      "LAN_ContaFinanceira": "String",
      "LAN_Historico": "String",
      "LAN_CRD_Desconto": "String",
      "LAN_CRD_Juros": "String",
      "LAN_CRD_Multa": "String",
      "IDWS": "String"
    }
  ]
}

```

6.1.3. Baixa do Contas a Receber Cartão

```

{
  "ArrayBaixaContasaReceberCartao": [
    {
      "CRE_Codigo": "String",
      "SequencialVencimento": "Integer",
      "Sequencial": "Integer",
      "Data": "TDateTime",
      "Valor": "Double",

```

```

        "Taxa": "Double",
        "DtCancel": "TDateTime",
        "MotivoCancel": "String",
        "LAN_ContaFinanceira": "String",
        "LAN_Historico": "String",
        "LAN_CRD_Taxa": "String",
        "IDWS": "String"
    }
}

```

6.1.4. Baixa do Contas a Receber Cheque

```

{
    "ArrayBaixaContasaReceberCheque": [
        {
            "CRE_Codigo": "String",
            "SequencialVencimento": "Integer",
            "Sequencial": "Integer",
            "Data": "TDateTime",
            "Valor": "Double",
            "Desconto": "Double",
            "Juros": "Double",
            "DtCancel": "TDateTime",
            "MotivoCancel": "String",
            "LAN_ContaFinanceira": "String",
            "LAN_Historico": "String",
            "LAN_CRD_Desconto": "String",
            "LAN_CRD_Juros": "String",
            "IDWS": "String"
        }
    ]
}

```

6.1.5. Cancela Baixa de Vencimentos

```

{
    "ArrayCancelaBaixaVencimento": [
        {
            "DtCancel": "TDateTime",
            "MotivoCancel": "String",
            "IDWS": "String"
        }
    ]
}

```

6.2. Métodos

6.2.1. Contas a Receber

- **IncluirContasaReceberComJSON**

- Assinatura:

```
function IncluirContasaReceberComJSON(const JSON: String):
String; stdcall;
```

- Parâmetros:

- Json do objeto TContasaReceber;

- Possíveis retornos:

- string no formato JSON;
- **AlterarContasaReceberComJSON**
 - Assinatura:


```
function AlterarContasaReceberComJSON(const JSON: String): String; stdcall;
```
 - Parâmetros:
 - Json do objeto TContasaReceber;
 - Possíveis retornos:
 - string no formato JSON;
- **IncluirListaContasaReceberComJSON**
 - Assinatura:


```
function IncluirListaContasaReceberComJSON(const ListaJSON: String): String; stdcall;
```
 - Parâmetros:
 - Json do objeto TAgendamentoContasaReceber (é um array de objetos do tipo TContasaReceber);
 - Possíveis retornos:
 - string no formato JSON;
- **AlterarListaContasaReceberComJSON**
 - Assinatura:


```
function AlterarListaContasaReceberComJSON(const ListaJSON: String): String; stdcall;
```
 - Parâmetros:
 - Json do objeto TAgendamentoContasaReceber (é um array de objetos do tipo TContasaReceber);
 - Possíveis retornos:
 - string no formato JSON;
- **IncluiListaBaixaContasReceberJSON**
 - Assinatura:


```
function IncluiListaBaixaContasReceberJSON(const ListaJSON: String): String; stdcall;
```
 - Parâmetros:
 - Json do objeto TAgendamentoBaixaContasaReceber (é um array de objetos do tipo TBaixaContasaReceber);
 - Possíveis retornos:
 - string no formato JSON;
- **IncluiListaBaixaContasReceberChequeJSON**
 - Assinatura:


```
function IncluiListaBaixaContasReceberChequeJSON(const ListaJSON: String): String; stdcall;
```
 - Parâmetros:
 - Json do objeto TAgendamentoBaixaContasReceberCheque (é um array de objetos do tipo TBaixaVencimentoChequeaReceber);
 - Possíveis retornos:

- string no formato JSON;
- **IncluiListaBaixaContasReceberCartaoJSON**
 - Assinatura:


```
function IncluiListaBaixaContasReceberCartaoJSON(const ListaJSON: String): String; stdcall;
```
 - Parâmetros:
 - Json do objeto TAgendamentoBaixaContasReceberCartao (é um array de objetos do tipo TBaixaVencimentoCartaoaReceber);
 - Possíveis retornos:
 - string no formato JSON;
- **GetJsonContasAReceberPorCodigo**
 - Assinatura:


```
function GetJsonContasAReceberPorCodigo(const Codigo: String): String; stdcall;
```
 - Parâmetros:
 - Código do Contas a Receber;
 - Possíveis retornos:
 - string do Json do objeto TContasaReceber, caso a busca tenha sido bem sucedida;
 - mensagem informando que o contas a receber não foi encontrado;
- **CancelaListaBaixaContasReceberJSON**
 - Assinatura:


```
function CancelaListaBaixaContasReceberJSON(const ListaJSON: String): String; stdcall;
```
 - Parâmetros:
 - Json do objeto TAgendamentoCancelaBaixaVencimento (é um array de objetos do tipo TCancelaBaixaVencimento);
 - Possíveis retornos:
 - string no formato JSON;
- **CancelaListaBaixaContasReceberCartaoJSON**
 - Assinatura:


```
function CancelaListaBaixaContasReceberCartaoJSON(const ListaJSON: String): String; stdcall;
```
 - Parâmetros:
 - Json do objeto TAgendamentoCancelaBaixaVencimento (é um array de objetos do tipo TCancelaBaixaVencimento);
 - Possíveis retornos:
 - string no formato JSON;
- **CancelaListaBaixaContasReceberChequeJSON**
 - Assinatura:


```
function CancelaListaBaixaContasReceberChequeJSON(const ListaJSON: String): String; stdcall;
```
 - Parâmetros:
 - Json do objeto TAgendamentoCancelaBaixaVencimento (é um array de objetos do tipo TCancelaBaixaVencimento);

- Possíveis retornos:
 - string no formato JSON;
- **CancelaListaVencimentosDiversosEmAbertoJSON**
 - Assinatura:


```
function CancelaListaVencimentosDiversosEmAbertoJSON(const
ListaJSON: String):String; stdcall;
```
 - Parâmetros:
 - Json do objeto TAgendamentoCancelaVencimento (é um array de objetos do tipo TCancelaVencimento);
 - Possíveis retornos:
 - string no formato JSON;
- **CancelaListaVencimentosCartaoaReceberEmAbertoJSON**
 - Assinatura:


```
function CancelaListaVencimentosCartaoaReceberEmAbertoJSON(const
ListaJSON: String):String; stdcall;
```
 - Parâmetros:
 - Json do objeto TAgendamentoCancelaVencimento (é um array de objetos do tipo TCancelaVencimento);
 - Possíveis retornos:
 - string no formato JSON;
- **CancelaListaVencimentosChequeaReceberEmAbertoJSON**
 - Assinatura:


```
function CancelaListaVencimentosChequeaReceberEmAbertoJSON(const
ListaJSON: String):String; stdcall;
```
 - Parâmetros:
 - Json do objeto TAgendamentoCancelaVencimento (é um array de objetos do tipo TCancelaVencimento);
 - Possíveis retornos:
 - string no formato JSON;
- **AlteraDataVencimentoDiversoEmAbertoJSON**
 - Assinatura:


```
function AlteraDataVencimentoDiversoEmAbertoJSON(const Json:
String): String; stdcall;
```
 - Parâmetros:
 - Json de Contas a Receber;
 - Possíveis retornos:
 - string no formato JSON;
- **AlteraListaDataVencimentoDiversoEmAbertoJSON**
 - Assinatura:


```
function AlteraListaDataVencimentoDiversoEmAbertoJSON(const Json:
String): String; stdcall;
```
 - Parâmetros:
 - Lista de Json de Contas a Receber;

- Possíveis retornos:
 - string no formato JSON;
- **EnviaCREContabilidadeComJSON**
 - Assinatura:


```
function EnviaCREContabilidadeComJSON(const Json: String):
String; stdcall;
```
 - Parâmetros:
 - Json. Ex.: {"codigocre": "20170700001", "idwscre": "12548"}
 - Possíveis retornos:
 - string no formato JSON;
- **EnviaListaCREContabilidadeComJSON**
 - Assinatura:


```
function EnviaListaCREContabilidadeComJSON(const Json: String):
String; stdcall;
```
 - Parâmetros:
 - Lista de Json .Ex.:


```
[{"codigocre": "20170700001", "idwscre": "12548"},
{"codigocre": "20170700002", "idwscre": "12548"}]
```
 - Possíveis retornos:
 - string no formato JSON;

7. Transferências entre Contas

7.1. Estrutura do JSON

```
{
  "Data": "string",
  "Sequencial": integer,
  "EstOrigem": "string",
  "EstDestino": "string",
  "ConOrigem": "string",
  "ConDestino": "string",
  "Valor": double,
  "Historico": "string",
  "HisCodigo": "string",
  "UsuCodigo": "string"
}
```

7.2. Métodos

- **IncluirTransferenciaEntreContasComJSON**
 - Assinatura:


```
function IncluirTransferenciaEntreContasComJSON(const      JSON:String):
string; stdcall;
```
 - Parâmetros:
 - Json do objeto TTransferenciaEntreContas;
 - Possíveis retornos:
 - string no formato JSON;
- **AlterarTransferenciaEntreContasComJSON**

- Assinatura:


```
function AlterarTransferenciaEntreContasComJSON(const      JSON:String):
string; stdcall;
```
 - Parâmetros:
 - Json do objeto TTransferenciaEntreContas;
 - Possíveis retornos:
 - string no formato JSON;
- **ExcluirTransferenciaEntreContasComJSON**
 - Assinatura:


```
function ExcluirTransferenciaEntreContasComJSON(const      JSON:String):
string; stdcall;
```
 - Parâmetros:
 - Json do objeto TTransferenciaEntreContas;
 - Possíveis retornos:
 - 'OK', caso a exclusão seja bem sucedida;
 - 'Transferência não cadastrada', caso a transferência passada não seja encontrada;
 - Descrição do Erro, caso aconteça algum erro durante a exclusão;
 - **GetJsonTransferenciaEntreContas**
 - Assinatura:


```
function GetJsonTransferenciaEntreContas(const Data: String;      const
Sequencial: Integer): String; stdcall;
```
 - Parâmetros:
 - Data;
 - Sequencial;
 - Possíveis retornos:
 - Json do objeto TTransferenciaEntreContas, caso a transferência passada seja encontrada;
 - 'Transferência não cadastrada', caso a transferência passada não seja encontrada;

8. Adiantamento a Cliente [Não implementado]

Entre em contato com o desenvolvimento da Fortes Tecnologia caso precise desta implementação.

9. Nota Fiscal de Saída [Não implementado]

Entre em contato com o desenvolvimento da Fortes Tecnologia caso precise desta implementação.

10. Contas a Pagar

10.1. Estrutura do JSON

10.1.1. Contas a Pagar

```
{
  "ArrayContasaPagar": [
    {
      "Codigo": "string",
      "Estabelecimento": "string",
      "CentroResultados": "string",
      "Fornecedor": "string",
      "CodigoFornecedor": "string",
      "Despesa": "string",
      "Documento": "string",
      "TipoDocumento": "string",
      "Emissao": "string",
      "MesAnoComp": "string",
    }
  ]
}
```

```

    "DtEntrada": "string",
    "Valor": "Double",
    "ContaFinanceira": "string",
    "ExportaAC": "integer",
    "OBS": "string",
    "ISSValor": "Double",
    "ISSBase": "Double",
    "IRRFBase": "Double",
    "IRRFValorCPF": "Double",
    "IRRFValorCNPJ": "Double",
    "INSSValorCPF": "Double",
    "INSSValorCNPJ": "Double",
    "INSSBase": "Double",
    "PISCOFINSCSLLValor": "Double",
    "PISCOFINSCSLLBase": "Double",
    "Origem": "string",
    "IDWS": "String",
    "VencimentosaPagar": [
      {
        "Codigo": "String",
        "Sequencial": "Integer",
        "Vencimento": "string",
        "Valor": "Double",
        "Titulo": "String",
        "DtCancel": "string"
      }
    ]
  }
}

```

10.1.2. Baixas do Contas a Pagar

```

{
  "ArrayBaixaContasaPagar": [
    {
      "CPG_Codigo": "String",
      "DataVencimento": "String",
      "SequencialVencimento": "Integer",
      "Sequencial": "Integer",
      "Data": "String",
      "Valor": "Double",
      "Desconto": "Double",
      "Juros": "Double",
      "DtCancel": "String",
      "MotivoCancel": "String",
      "Historico": "String",
      "LAN_ContaFinanceira": "String",
      "LAN_Historico": "String",
      "LAN_CRD_Desconto": "String",
      "LAN_CRD_Juros": "String",
      "IDWS": "String"
    }
  ]
}

```

10.2. Métodos

- **IncluirListaContasaPagarComJSON**

- Assinatura:

```

function IncluirListaContasaPagarComJSON(const ListaJSON:
string): string; stdcall;

```


- Parâmetros:
 - Json do objeto TAgendamentoContasaPagar (é um array de objetos do tipo TContasaPagar);
- Possíveis retornos:
 - string no formato JSON;
- **IncluirListaBaixaContasaPagarJSON**
 - Assinatura:


```
function IncluirListaBaixaContasaPagarJSON(const ListaJSON: string): string; stdcall;
```
 - Parâmetros:
 - Json do objeto TAgendamentoBaixaContasaPagar (é um array de objetos do tipo TBaixaVencimentoaPagar);
 - Possíveis retornos:
 - string no formato JSON;
- **CancelarListaBaixaContasaPagarJSON**
 - Assinatura:


```
function CancelarListaBaixaContasaPagarJSON(const ListaJSON: string): string; stdcall;
```
 - Parâmetros:
 - Json do objeto TAgendamentoCancelaBaixaVencimento (é um array de objetos do tipo TCancelaBaixaVencimento);
 - Possíveis retornos:
 - string no formato JSON;
- **CancelaListaVencimentosaPagarEmAbertoJSON**
 - Assinatura:


```
function CancelaListaVencimentosaPagarEmAbertoJSON(const ListaJSON: string): string; stdcall;
```
 - Parâmetros:
 - Json do objeto TAgendamentoCancelaVencimento (é um array de objetos do tipo TCancelaVencimento);
 - Possíveis retornos:
 - string no formato JSON;
- **GetMovimentacaoBaixaAPagar**
 - Assinatura:


```
GetMovimentacaoBaixaAPagar(const EST_Codigo: WideString; const DataInicial: WideString; const DataFinal: WideString; const Cancelado: Boolean): TlistaMovimentacaoBaixaAPagar; stdcall;
```
 - Parâmetros:
 - Estabelecimento (vazio("")) considera todos;
 - Período (data inicial e final no formato DD/MM/YYYY);
 - Cancelados (true) ou não (false);
 - Possíveis retornos:
 - Array vazia ("), caso não exista baixa conforme os parâmetros passados;
 - Lista de baixas efetuadas no Array;

11. Adiantamento a Fornecedor [Não implementado]

Entre em contato com o desenvolvimento da Fortes Tecnologia caso precise desta implementação.

12. Requisições [Não implementado]

Entre em contato com o desenvolvimento da Fortes Tecnologia caso precise desta implementação.

13. Entradas [Não implementado]

Entre em contato com o desenvolvimento da Fortes Tecnologia caso precise desta implementação.

14. Saídas

- **IncluiSaidasJSON**

- Assinatura:

- ```
function IncluiSaidasJSON(const JSON: String): String;
stdcall;
```

- Parâmetros:

- String no formato JSON ;

- Possíveis retornos:

- string no formato JSON, com o código da saída incluída, caso a inclusão tenha sido realizada com sucesso;
    - string no formato JSON, com a descrição do Erro, caso aconteça algum erro durante a inclusão;

Exemplo:

```
{
 "Estabelecimento": "String",
 "Requisitante": "String",
 "CentroDeEstoque": "String",
 "CentroDeResultados": "String",
 "CentroDeResultadosOrigem": "String",
 "Documento": "String",
 "Data": "String formato DD/MM/AAAA",
 "Hora": "String formato HH:MM:SS",
 "Obs": "String",
 "IDWS": "String",
 "ItensSaida": [{
 "Item": "String",
 "Sequencial": "integer",
 "Quantidade": Double,
 "LOT_Sequencial": "integer",
 "ContaAplicacao": "String"
 }]
}
```

- **IncluiListaSaidasJSON**

- Assinatura:

- ```
function IncluiListaSaidasJSON(const JSON: String): String;
stdcall;
```

- Parâmetros:

- Lista de strings no formato JSON;

- Possíveis retornos:

- string no formato JSON, com o código da saída incluída, caso a inclusão tenha sido realizada com sucesso;
- string no formato JSON, com a descrição do Erro, caso aconteça algum erro durante a inclusão;

Exemplo:

```
[{
  "Estabelecimento": "String",
  "Requisitante": "String",
  "CentroDeEstoque": "String",
  "CentroDeResultados": "String",
  "CentroDeResultadosOrigem": "String",
  "Documento": "String",
  "Data": "String formato DD/MM/AAAA",
  "Hora": "String formato HH:MM:SS",
  "Obs": "String",
  "IDWS": "String",
  "ItensSaida": [{
    "Item": "String",
    "Sequencial": "integer",
    "Quantidade": Double,
    "LOT_Sequencial": "integer",
    "ContaAplicacao": "String"
  }]
},
{
  "Estabelecimento": "String",
  "Requisitante": "String",
  "CentroDeEstoque": "String",
  "CentroDeResultados": "String",
  "CentroDeResultadosOrigem": "String",
  "Documento": "String",
  "Data": "String formato DD/MM/AAAA",
  "Hora": "String formato HH:MM:SS",
  "Obs": "String",
  "IDWS": "String",
  "ItensSaida": [{
    "Item": "String",
    "Sequencial": "integer",
    "Quantidade": Double,
    "LOT_Sequencial": "integer",
    "ContaAplicacao": "String"
  }]
}]
```

15. Devoluções de Saída [Não implementado]

Entre em contato com o desenvolvimento da Fortes Tecnologia caso precise desta implementação.

16. Pedido de Venda [Não implementado]

Entre em contato com o desenvolvimento da Fortes Tecnologia caso precise desta implementação.

17. Receitas e Despesas

17.1. Estrutura do JSON

```
{
  "ReceitaDespesa":
  [
    {
      "CODIGO": "string",
      "NOME": "string",
      "NATUREZA": "string"
    }
  ]
}
```

17.2. Métodos

- **IncluiListaReceitaDespesaJSON**
 - Assinatura:
`function IncluiListaReceitaDespesaJSON(const ListaJSON: String): String; stdcall;`
 - Parâmetros:
 - Json do objeto TAgendamentoReceitasDespesas (é um array de objetos do tipo TReceitasDespesas);
 - Possíveis retornos:
 - string no formato JSON;

Verificando a informação no Fortes Financeiro

Caso o Fortes Financeiro esteja instalado, é possível verificar as operações realizadas via *webservice* consultando o próprio sistema. Se você não tem muita familiaridade com o mesmo, pode-se utilizar a opção “CTRL + L” [Figura 17] e [Figura 18] para encontrar as funcionalidades desejadas.

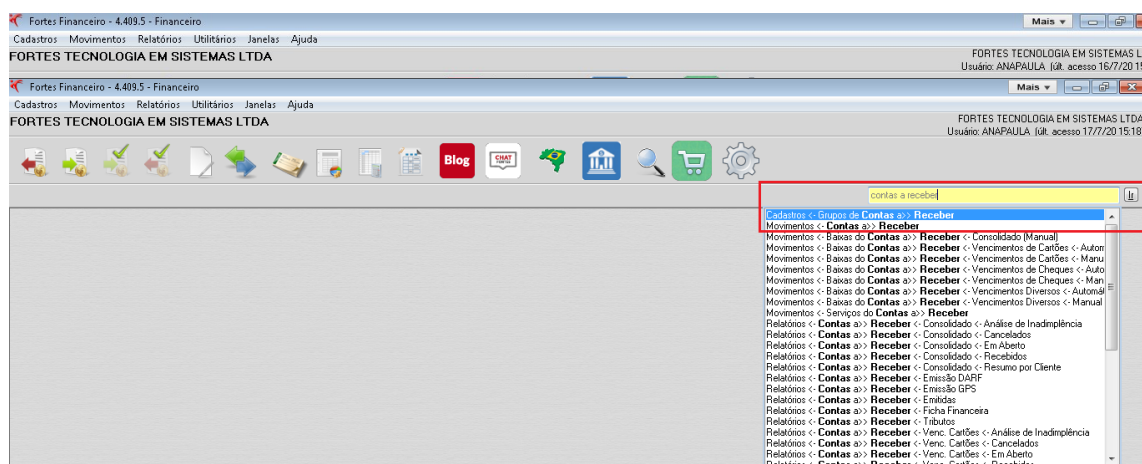


Figura 17

Figura 18

Caso o Fortes Financeiro não esteja instalado, pode-se verificar as operações realizadas diretamente

no banco de dados. A seguir a lista de tabelas com suas descrições.

Tabela	Descrição
CLI	Tabela de Cliente
SCL	Tabela de Serviço de Cliente
FOR	Tabela de Fornecedor

SER	Tabela de Serviço
PRO	Tabela de Produto (Item)
CRE	Tabela de Contas a Receber
VDR	Tabela de Vencimentos Diversos do Cartão do Contas a Receber
VCR	Tabela de Vencimentos do Cartão do Contas a Receber
VHR	Tabela de Vencimentos de Cheques do Contas a Receber
RRR	Tabela de Rateio das Receitas do Contas a Receber
ICR	Tabela de Itens do Contas a Receber
CIR	Tabela de Comissões dos Itens do Contas a Receber
SCR	Tabela de Serviços do Contas a Receber
CSR	Tabela de Comissões dos Serviços do Contas a Receber
BVC	Tabela de Baixas de Vencimentos de Cartões do Contas a Receber
RRR	Tabela de Rateio das Receitas do Contas a Receber
LAN	Tabela de Lançamentos
TCF	Tabela de Transferência entre Contas
ACL	Tabela de Adiantamento a clientes
NFS	Tabela de Nota Fiscal de Saída
SNF	Tabela de Serviços da Nota Fiscal
PNF	Tabela de Produtos (Itens) da Nota Fiscal
DNF	Tabela de Documentos referenciados na Nota Fiscal
FPN	Tabela de Formas de Pagamento da Nota Fiscal
CPG	Tabela de Contas a Pagar
VCP	Tabela de Vencimento do Contas a Pagar
DCP	Tabela de Despesas do Contas a Pagar
BVP	Tabela de Baixas de Vencimentos a Pagar
CPD	Tabela de Documentos do Contas a Pagar
VIC	Tabela de Vencimento dos Impostos do Contas a Pagar
ACL	Tabela de Adiantamento a Cliente
AFR	Tabela de Adiantamento a Fornecedor
RIT	Tabela de Requisição de Itens
ITE	Tabela de Itens da Requisição
ENT	Tabela de Entrada
IEN	Tabela de Itens da Entrada
SIT	Tabela de Saída
ISI	Tabela de Itens da Saída
PEV	Tabela de Pedido de Venda
IPV	Tabela de Item Pedido de Venda
DIC	Tabela de Devolução de Saída
IDI	Tabela de Itens da Devolução de Saída
CRD	Tabela de Receitas/Despesas
CRS	Tabela de Centro de Resultados

Tabelas Auxiliares

Tabela 1 – Finalidades DOC

Código	Descrição
00	Não se Aplica
01	Crédito em Conta
02	Pagamento de Aluguel/Condomínio
03	Pagamento de Duplicata/Títulos
04	Pagamento de Dividendos
05	Pagamento de Mensalidade Escolar
06	Pagamento de Salários
07	Pagamento de Fornecedores/Honorários
08	Operações de Câmbios/Fundos/Bolsa de Valores
09	Repasse de Arrecadação/Pagamento de Tributos
10	Transferência Internacional em Real
11	DOC para Poupança

Tabela 2 - Leiautes FEBRABAN

Código	Descrição
1	FEBRABAN - 1
2	FEBRABAN - 2
3	FEBRABAN - 3
4	FEBRABAN - 4
5	FEBRABAN - 5
6	FEBRABAN - 6
7	FEBRABAN - 7
9	SICAP

Tabela 3 - Tipos de Fatura

Código	Descrição
N	Não informado
V	À vista
P	À prazo

Tabela 4 - Código Situação Tributária coluna A (CSTA)

Código	Descrição
0	Nacional
1	Estrangeira (Importação Direta)
2	Estrangeira (Adquirida no Mercado Interno)

3	Nacional, mercadoria ou bem com Conteúdo de Importação a 40%(quarenta por cento)
4	Nacional, cuja produção tenha sido feita em conformidade com os processos produtivos básicos
5	Nacional, mercadoria ou bem com Conteúdo de Importação inferior ou igual a 40% (quarenta por cento)
6	Estrangeira - Importação direta, sem similar nacional, constante em lista de Resolução CAMEX
7	Estrangeira - Adquirida no mercado interno, sem similar nacional, constante em lista de Resolução CAMEX

Tabela 5 - Código Situação Tributária coluna B (CSTB)

Código	Descrição
00	Tributada Integralmente
10	Tributada e com Cobrança de ICMS Substituição Tributária
20	Tributada com Redução de Base de Cálculo
30	Isenta ou Não Tributada e com Cobrança de ICMS Substituição Tributária
40	Isenta
41	Não Tributada
50	Suspensão
51	Diferimento
60	ICMS Cobrado Anteriormente por Substituição Tributária
70	Com Redução de Base de Cálculo e Cobrança do ICMS por Substituição Tributária
90	Outras

Tabela 6 - Código de Situação da Operação no Simples Nacional (CSOSN)

Código	Descrição
101	Tributada pelo Simples Nacional com permissão de crédito
102	Tributada pelo Simples Nacional sem permissão de crédito
103	Isenção do ICMS no Simples Nacional para faixa de receita bruta
201	Tributada pelo Simples Nacional com permissão de crédito e com cobrança do ICMS por substituição tributária
202	Tributada pelo Simples Nacional sem permissão de crédito e com cobrança do ICMS por substituição tributária
203	Isenção do ICMS no Simples Nacional para faixa de receita bruta e com cobrança do ICMS por substituição tributária
300	Imune
400	Não tributada pelo Simples Nacional
500	ICMS cobrado anteriormente por substituição tributária (substituído) ou por antecipação
900	Outros

Tabela 7 - Tipo Tributação ISS

Município	Código	Descrição
Fortaleza (CE)	1	Normal
	2	Estimativa (DDS - Fortaleza(CE))
	3	Isenta
	4	Não-Incidência
	5	Imune
	6	ISS Devido a Outro Município (DDS - Fortaleza (CE))
	7	Optante do Simples Nacional
	8	Exigibilidade Suspensa por Decisão Judicial
	9	Exigibilidade Suspensa por Procedimento Administrativo
Caucaia (CE)	1	Normal
	2	Estimativa
	3	Isenta
	4	Não-Incidência
	5	Imune
	6	ISS Devido a Outro Município
	7	Optante do Simples Nacional
Iguatu (CE)	1	Normal
	2	Estimativa
	3	Isento de ISSQN
	4	Não-Incidência
	5	Imune
	6	Não Tributável
Teresina (PI) São Luiz (MA) Salvador (BA) Feira de Santana (BA) Sobral (CE) Belo Horizonte (MG) Municípios do Rio Grande do Norte Manaus (AM)	1	Normal
	2	Estimativa
	3	Isenta
	4	Não-Incidência
	5	Imune
	7	Tributada Simples Nacional
Demais Municípios	1	Normal
	2	Estimativa
	3	Isenta
	4	Não-Incidência
	5	Imune

Tabela 8 - Código da Situação Tributária do COFINS/PIS (Tabela 18 do AC Fiscal)

Código	Descrição
--------	-----------

01	Operação Tributável - Base de Cálculo = Valor da Operação Alíquota Normal (Cumulativo/Não Cumulativo)
02	Operação Tributável - Base de Calculo = Valor da Operação (Alíquota Diferenciada)
03	Operação Tributável - Base de Calculo = Quantidade Vendida x Alíquota por Unidade de Produto
04	Operação Tributável - Tributação Monofásica - (Alíquota Zero)
05	Operação Tributável - (Substituição Tributária)
06	Operação Tributável - Alíquota Zero
07	Operação Isenta da Contribuição
08	Operação Sem Incidência da Contribuição
09	Operação com Suspensão da Contribuição
49	Outras Operações de Saída
50	Operação com Direito a Crédito - Vinculada Exclusivamente a Receita Tributada no Mercado Interno
51	Operação com Direito a Crédito - Vinculada Exclusivamente a Receita Não Tributada no Mercado Interno
52	Operação com Direito a Crédito - Vinculada Exclusivamente a Receita de Exportação
53	Operação com Direito a Crédito - Vinculada a Receitas Tributadas e Não Tributadas no Mercado Interno
54	Operação com Direito a Crédito - Vinculada a Receitas Tributadas no Mercado Interno e de Exportação
55	Operação com Direito a Crédito - Vinculada a Receitas Não-Tributadas no Mercado Interno e de Exportação
56	Operação com Direito a Crédito - Vinculada a Receitas Tributadas e Não Tributadas no Mercado Interno, e de Exportação
60	Crédito Presumido - Operação de Aquisição Vinculada Exclusivamente a Receita Tributada no Mercado Interno
61	Crédito Presumido - Operação de Aquisição Vinculada Exclusivamente a Receita Não-Tributada no Mercado Interno
62	Crédito Presumido - Operação de Aquisição Vinculada Exclusivamente a Receita de Exportação
63	Crédito Presumido - Operação de Aquisição Vinculada a Receitas Tributadas e Não-Tributadas no Mercado Interno
64	Crédito Presumido - Operação de Aquisição Vinculada a Receitas Tributadas no Mercado Interno e de Exportação
65	Crédito Presumido - Operação de Aquisição Vinculada a Receitas Não-Tributadas no Mercado Interno e de Exportação
66	Crédito Presumido - Operação de Aquisição Vinculada a Receitas Tributadas e Não-Tributadas no Mercado Interno e de Exportação
67	Crédito Presumido - Outras Operações
70	Operação de Aquisição sem Direito a Crédito

71	Operação de Aquisição com Isenção
72	Operação de Aquisição com Suspensão
73	Operação de Aquisição a Alíquota Zero
74	Operação de Aquisição sem Incidência da Contribuição
75	Operação de Aquisição por Substituição Tributária
98	Outras Operações de Entrada
99	Outras Operações

Tabela 9 - Tipo Tributação ICMS/IPI (Tabela 13 do layout do AC Fiscal)

Código	Descrição
1	Com Débito/Crédito
2	Isentas/Não-Tributadas
3	Outras

Tabela 10 - Espécie dos Documentos (Tabela 10 do Layout do AC Fiscal)

Código	Descrição
ACT	Autorização de Carregamento de Transportes
BPA	Bilhete de Passagem Aquaviária
BPF	Bilhete de Passagem Ferroviário
BPNB	Bilhete de Passagem e Nota de Bagagem
BPR	Bilhete de Passagem Rodoviário
CA	Conhecimento Aéreo
CF	Cupom Fiscal
CTA	Conhecimento de Transporte Avulso
CTAC	Conhecimento de Transporte Aquaviário de Cargas
CTFC	Conhecimento de Transporte Ferroviário de Cargas
CTRC	Conhecimento de Transporte Rodoviário de Cargas
CTE	Conhecimento de Transporte Eletrônico
DT	Despacho de Transporte
MC	manifesto de Cargas
MRE	Mapa Resumo ECF
NF1	Nota Fiscal - Modelo 1
NF1A	Nota Fiscal - Modelo 1ª
NFA	Nota Fiscal Avulsa
NFE	Nota Fiscal Eletrônica
NFCEE	NotaFiscal Eletrônica
NFP	Nota Fiscal De Produtor
NFF	Nota Fiscal Fatura

NFSC	Nota Fiscal de Serviço de Comunicação
NFST	Nota Fiscal de Serviço de Transporte
NFSTC	Nota Fiscal de Serviço de Telecomunicação
NFVC	Nota Fiscal de Venda a Consumidor
OCC	Ordem de COleta de Cargas
RMD	Resumo de Movimento Diário
NFCE	Nota Fiscal de Venda para Consumidor Final Eletrônica

Tabela 11 - Situação dos Documentos (Tabela 26 do layout do Fiscal)

Código	Descrição
0	Normal
1	Cancelado
4	NFE denegada
5	NFe numeração inutilizada
8	Documento Fiscal emitido com base em Regime Especial ou Norma Específica

Tabela 12 - Unidades de Medida (Tabela 15 do layout do AC Fiscal)

Código	Descrição
0	Não informada
1	Peso (kg)
2	Litro (l)
3	Volume (m3)

Tabela 13 - Tipo de Cálculo do Frete (Tabela 14 do layout do AC Fiscal)

Código	Descrição
0	Não informado
1	Pelo peso das mercadorias
2	Pelo valor das mercadorias
3	Pelo peso e valor das mercadorias
4	Outras formas de cálculo

Tabela 14 - Natureza do Frete (Tabela 28 do layout do AC Fiscal)

Código	Descrição
0	Operações de vendas, com ônus suportado pelo estabelecimento vendedor
1	Operações de vendas, com ônus suportado pelo adquirente
2	Operações de compras (bens para revenda, matérias-primas e outros produtos,

	geradores de crédito)
3	Operações de compras (bens para revenda, matérias-primas e outros produtos, não geradores de crédito)
4	Transferência de produtos acabados entre estabelecimentos da pessoa jurídica
5	Transferência de produtos em elaboração entre estabelecimentos da pessoa jurídica
9	Outras

Tabela 15 - Responsável Frete (Tabela 11 do layout do AC Fiscal)

Código	Descrição
N	Não Informado
R	Remetente
D	Destinatário

Tabela 16 - Natureza do Crédito (Tabela 29 do layout do AC Fiscal)

Código	Descrição
01	Aquisição de bens para revenda
02	Aquisição de bens utilizados como insumo
03	Aquisição de serviços utilizados como insumo
04	Energia elétrica e térmica, inclusive sob a forma de vapor
05	Aluguéis de prédios
06	Aluguéis de máquinas e equipamentos
07	Armazenagem de mercadoria e frete na operação de venda
08	Contraprestações de arrendamento mercantil
09	Máquinas, equipamentos e outros bens incorporados ao ativo imobilizado (crédito sobre encargos de depreciação)
10	Máquinas, equipamentos e outros bens incorporados ao ativo imobilizado (crédito com base no valor de aquisição)
11	Amortização e Depreciação de edificações e benfeitorias em imóveis
12	Devolução de Vendas Sujeitas à Incidência Não-Cumulativa
13	Outras Operações com Direito a Crédito
14	Atividade de Transporte de Cargas - Subcontratação
15	Atividade Imobiliária - Custo Incorrido de Unidade Imobiliária
16	Atividade Imobiliária - Custo Orçado de unidade não concluída
17	Atividade de Prestação de Serviços de Limpeza, Conservação e Manutenção - vale-transporte, vale-refeição ou vale-alimentação, fardamento ou uniforme.
18	Estoque de abertura de bens

Tabela 17 - Origem do Contas a Pagar

Código	Descrição
--------	-----------

O	Outro
N	INSS
S	ISS
R	IRRF
P	PIS/COFINS/CSLL
C	Cargas
M	Comissão
F	Frota
I	ICMS

Tabela 18 - Motivo do Inventário (Tabela 34 do layout do AC Fiscal)

Código	Descrição
01	No final do período
02	Na mudança de forma de tributação da mercadoria (ICMS)
03	Na solicitação da baixa cadastral, paralisação temporária e outras situações
04	Na alteração de regime de pagamento - condição do contribuinte
05	Por determinação dos fiscos

Tabela 19 - Tipo de Tributação do ICMS (Tabela 16 do layout do Fiscal)

Código	Descrição
0	Com débito
1	Substituição
2	ISENTAS
3	Não Incidência

Tabela 20 - Situação Tributária do IPI

Código	Descrição
00	Entrada com Recuperação de Crédito
01	Entrada Tributável com Alíquota Zero
02	Entrada Isenta
03	Entrada Não-Tributada
04	Entrada Imune
05	Entrada com Suspensão
49	Outras Entradas
50	Saída Tributada
51	Saída Tributável com Alíquota Zero
52	Saída Isenta

53	Saída Não-Tributada
54	Saída Imune
55	Saída com Suspensão
99	Outras Saídas