



SECRETARIA DA FAZENDA
Governo do Estado do Ceará

PROJETO MFE CF-e
MÓDULO FISCAL ELETRÔNICO
CU POM FISCAL ELETRÔNICO



SISTEMA DE AUTENTICAÇÃO E TRANSMISSÃO
DE DOCUMENTOS FISCAIS ELETRÔNICOS

Manual do DRIVER MFE

Biblioteca de Funções

Comunicador

Monitor

Data de Criação: 05 de outubro de 2015

Última Atualização: 05 de agosto de 2016.

Versão: Manual Driver MFE 1.0.4



CONTROLE DE VERSÕES

Versão	Data	Observações
1.0.0	05/10/2015	Versão inicial
1.0.1	10/11/2015	Revisão do instalador
1.0.2	27/04/2016	Revisão geral
1.0.3	22/07/2016	Alteração do ícone do Monitor; inclusão das versões para Linux; ajustes na ilustração "Componentes do PDV"
1.0.4	05/08/2016	Inclusão do capítulo para instalação em Linux



DEFINIÇÕES E SIGLAS

Termo	Descrição
AC	Aplicativo Comercial Aplicativo para gestão das vendas e solicitação de emissão de Cupons Fiscais de Venda e de Cancelamento. Deve ser compatível com o MFE.
AC SEFAZ-CE	Autoridade Certificadora que gerencia (emite e revoga) certificados digitais de equipamentos MFE (e-SAT) reconhecidos pela ICP-Brasil e providos pela própria SEFAZ-CE.
CFe	Cupom Fiscal Eletrônico de Venda ou de Cancelamento.
Configurador	Aplicativo para configuração e ativação do MFE.
MFE	Módulo Fiscal Eletrônico Equipamento adotado pelo Estado do Ceará para emissão de Cupons Fiscais eletrônicos, nos termos do Projeto CFe, dotado de recursos de hardware e software adicionais necessários ao modelo de negócios do Ceará.
PDV	Ponto do Venda Equipamento do Contribuinte onde está conectado o MFE e que realiza a emissão de CFes de venda ou de cancelamento.
Projeto CFe	Conjunto de especificações técnicas definidas pelos documentos de Especificação de Requisitos (ER), Manual de Orientação (MO) e demais documentos estabelecidos pela COTEPE conforme AJUSTE SINIEF nº 11 de 24 de Setembro de 2010 e posteriores alterações.
Projeto MFE	Projeto da SEFAZ-CE para implantação do Cupom Fiscal Eletrônico, baseado no Projeto CFE porém acrescido de funcionalidades específicas para o Estado do Ceará definidas pela SEFAZ-CE no documento de Especificação de Requisitos do MFE (ER-MFE) e demais documentos de apoio disponíveis no sítio eletrônico: HTTP://CFE.SEFAZ.CE.GOV.BR.
SEFAZ-CE	Secretaria da Fazenda do Estado do Ceará.



ÍNDICE

1	INTRODUÇÃO.....	1
1.1	Componentes do DRIVER MFE	1
1.2	Suporte Técnico.....	2
2	PONTO DE VENDA.....	3
2.1	Aplicativo Comercial	4
2.2	Driver MFE.....	4
2.3	Driver de comunicação USB-SERIAL	4
3	INSTALAÇÃO	5
3.1	Sistema Operacional Windows.....	5
3.2	Sistema Operacional Linux.....	8
4	COMUNICADOR	10
5	MONITOR.....	10
5.1	Ícones	11
5.2	Tela de Status.....	12
6	BIBLIOTECA DE FUNÇÕES	14
6.1	Funções do Projeto CFe.....	14
6.2	Funções do Projeto MFE	22
7	PROTÓTIPOS DAS FUNÇÕES	24



1 INTRODUÇÃO

Este documento destina-se ao pessoal técnico responsável pela instalação do PDV e/ou desenvolvimento de Aplicativos Comerciais e Módulos Fiscais MFE. Descreve o Driver MFE, sua instalação, seus componentes e as funções disponíveis.

Com o intuito de padronizar e facilitar o uso do MFE a SEFAZ-CE fornece um conjunto de componentes de software que devem ser instalados no equipamento do Contribuinte (PDV) e serão utilizados na comunicação com o MFE.

Este conjunto de componentes de software é chamado de Driver MFE e está disponível no sítio da SEFAZ-CE na forma de um programa instalador no seguinte endereço:

<http://cfe.sefaz.ce.gov.br>

Informações técnicas mais detalhadas poderão ser encontradas na **Especificação Técnica de Requisitos do MFE do Projeto MFE** disponível no sítio da SEFAZ-CE.

1.1 Componentes do DRIVER MFE

O Driver MFE é composto pelos seguintes componentes:

- Biblioteca de Funções
- Comunicador
- Monitor



1.2 Suporte Técnico

A SEFAZ-CE proverá suporte nos assuntos relacionados ao uso do DRIVER MFE e nos processos relativos à emissão de Cupom Fiscal eletrônico para os Contribuintes, desenvolvedores de Aplicativos Comerciais (Software Houses) e Fabricantes de equipamentos MFE, através dos seguintes meios:

- **Sítio eletrônico**
 - SEFAZ: <http://www.sefaz.ce.gov.br>
 - Projeto MFE: <http://cfe.sefaz.ce.gov.br>
- **Email**
 - Contribuintes: mfe@sefaz.ce.gov.br
 - Fabricantes de MFE e Software Houses: suporte.mfe@sefaz.ce.gov.br
- **Telefone**
 - Call Center: 0800-707-8585



2 PONTO DE VENDA

O Ponto de Venda, ou PDV, é o equipamento do contribuinte responsável pelo processo de emissão de Cupons Fiscais, através de um Aplicativo Comercial, e onde o MFE é conectado a uma porta USB.

Além do Aplicativo Comercial o PDV poderá também executar o Software de Ativação e Configuração fornecido pelo Fabricante do MFE.

No PDV serão instalados também componentes de software disponibilizados pela SEFAZ-CE necessários para a comunicação com o MFE e descritos adiante.

Para o funcionamento do PDV são necessários os seguintes componentes:

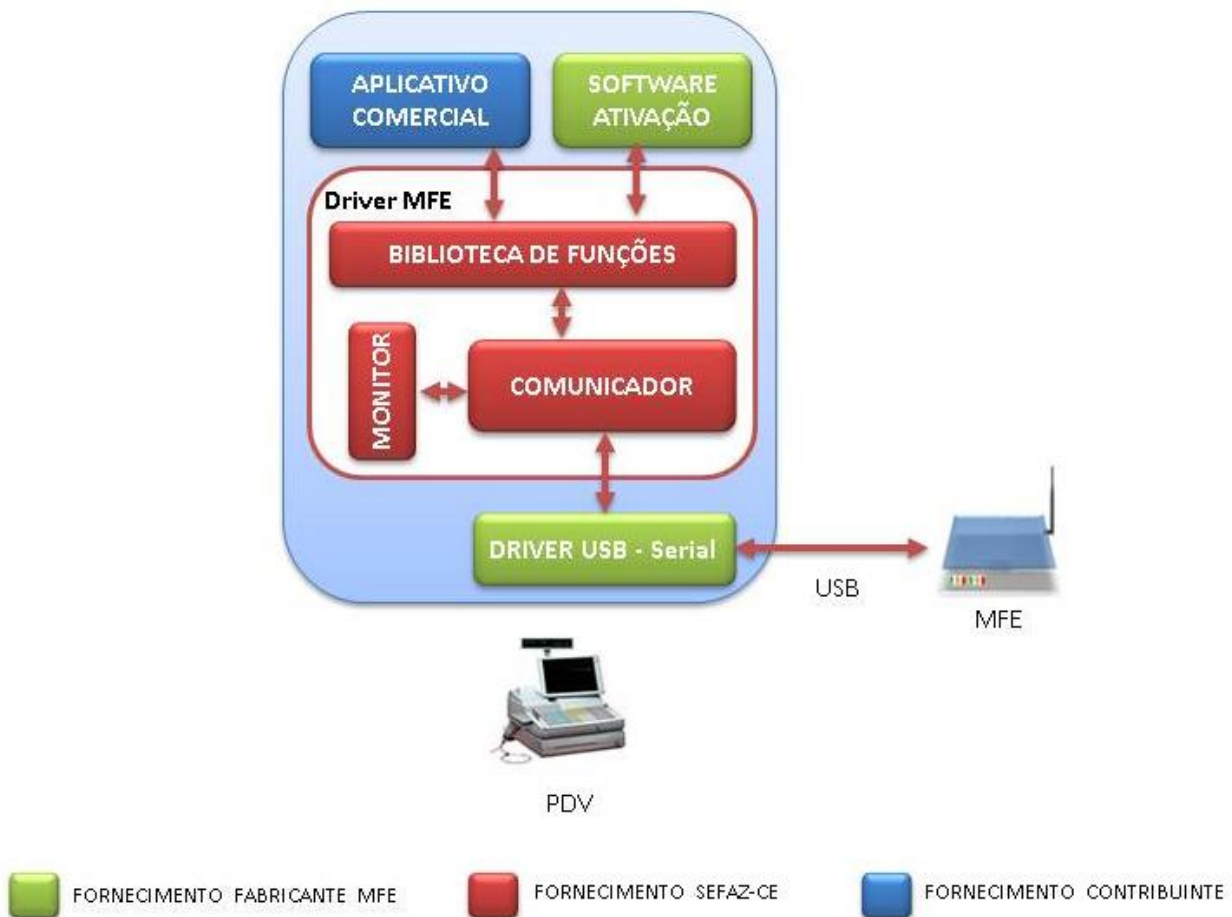


Figura 1 - Componente do Ponto de Venda

2.1 Aplicativo Comercial

É a aplicação responsável por enviar os dados de venda ou de cancelamento para o MFE para que ele possa gerar o respectivo Cupom Fiscal Eletrônico, além de permitir a execução de outras funcionalidades.

Poderá ser utilizado qualquer Aplicativo Comercial compatível com os padrões do MFE-CFe/SAT-CFe.

2.2 Driver MFE

Para facilitar a instalação e atualização dos componentes de software disponibilizados pela SEFAZ-CE, eles foram empacotados em uma única instalação chamada DRIVER MFE.

Os componentes do Driver MFE são:

Componente	Descrição
Biblioteca de Funções	Biblioteca de funções (arquivo com extensão “.DLL” no Windows ou “.SO” no Linux) que disponibiliza para as aplicações (AC/Software de Ativação) todas as funcionalidades necessárias para a operação do Módulo Fiscal MFE padronizando assim toda a comunicação. Em ambiente Windows 32 bits ela se chama mfe.dll e se encontra na pasta: C:\Windows\System32 Em ambiente Windows 64 bits ela se chama mfe.dll e se encontra na pasta: C:\Windows\SystemWOW64
Monitor	Aplicação que informa, através de ícones na barra de status, a disponibilidade ou não de um Módulo Fiscal MFE e permite a visualização de algumas informações.
Comunicador	É uma aplicação executada como um serviço responsável por realizar a comunicação com o Módulo Fiscal.

O Driver MFE é compatível com os seguintes sistemas operacionais:

- Windows XP, Vista, 7, 8, 8.1 e 10 (32 e 64 bits)
- Linux OpenSUSE 12. 2e 12.3, Debian 7.3 e 8.1, e Fedora 21 e 22 (versões 32 e 64 bits)

2.3 Driver de comunicação USB-SERIAL

Driver responsável por permitir a conversão do dispositivo USB, ao qual o Módulo Fiscal MFE está conectado, em interface serial conforme o sistema operacional do PDV e disponibilizá-la para uso pelo serviço Comunicador.

Exemplos de portas seriais: COM11 (Windows) ou ttyS1 (Linux).



3 INSTALAÇÃO

O programa instalador do DRIVER MFE instalará automaticamente todos os seus componentes e configurará o sistema operacional do PDV para executar os serviços e aplicações necessárias.

Antes da sua instalação deve-se:

1. Instalar o Driver de comunicação USB-SERIAL disponibilizado pelo Fabricante do MFE.
2. Confirmar a existência de uma nova porta serial disponível quando o MFE está conectado.

Observação: De acordo com as configurações de segurança e atualizações do Windows em uso, algumas janelas podem não ser exibidas por serem automaticamente aceitas pelo sistema, neste caso ignore o passo em questão.

3.1 Sistema Operacional Windows

A seguir são descritos os passos para a instalação do Driver MFE para o Sistema Operacional Windows:

1. Execute o arquivo de instalação do DRIVER MFE.
2. Caso o Windows exiba uma janela perguntando “**Deseja permitir que o programa de um fornecedor desconhecido faça alterações nesse computador**”, confirme a execução.
3. Será exibida a tela de boas vindas, clique no botão “**Avançar**”:

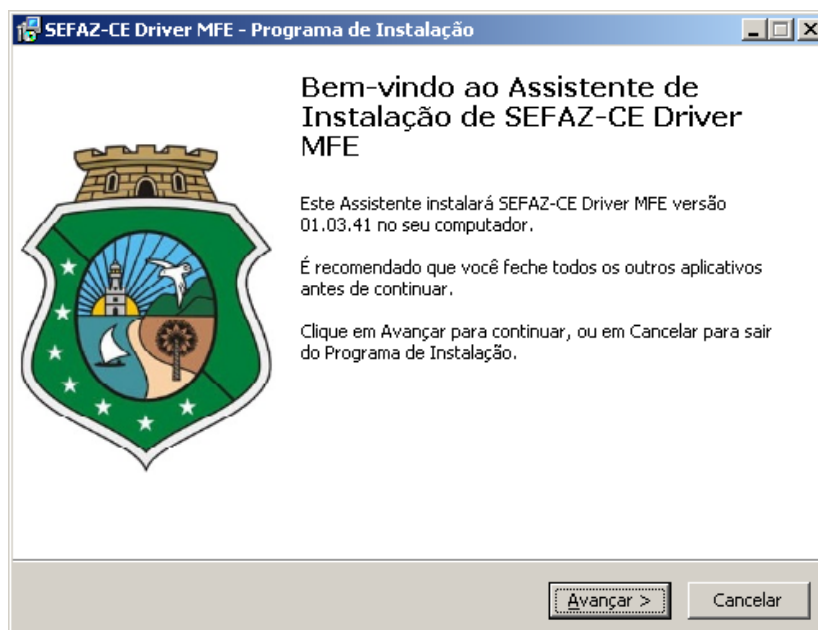


Figura 2 - Tela Inicial do Instalador

4. O instalador copiará os arquivos necessários conforme apresentado a figura abaixo:

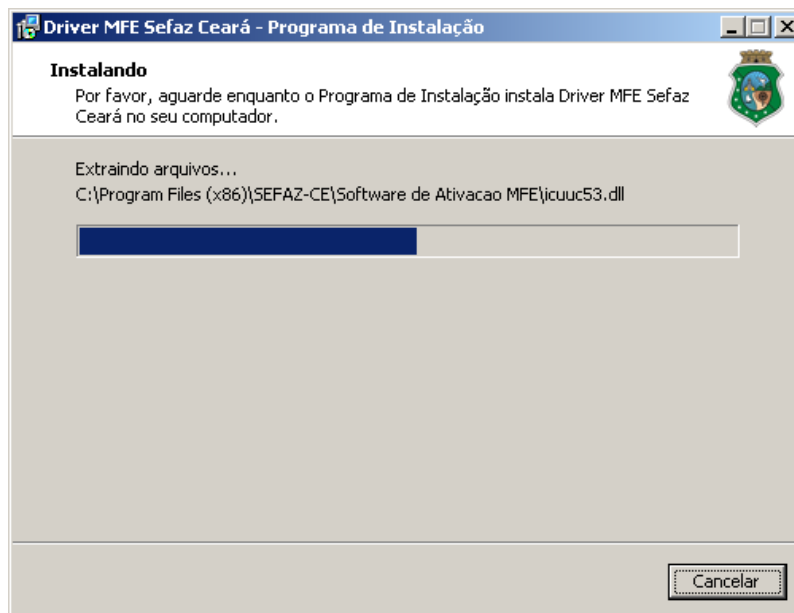


Figura 3 – Instalador copiando arquivos para o Computador

Observação: Alguns antivírus, tais como o Norton Security, podem reconhecer o Monitor do Driver MFE como um vírus. Neste caso poderá ser solicitada uma ação ao usuário para permitir que o Monitor execute.

Ao fim da instalação será executado automaticamente o Monitor MFE que aparecerá em sua barra de status.

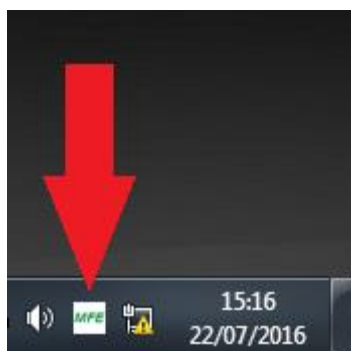


Figura 4 - Ícone do Monitor do Driver MFE na barra de status

Neste momento todos os softwares para uso do MFE foram instalados:

- A Biblioteca de Funções foi instalada
- O Monitor é mostrado na barra de Status



- O Serviço Comunicador já está em execução.

Observações:

1. De acordo com as configurações de segurança e atualizações do Windows em uso, algumas janelas podem não ser exibidas por serem automaticamente aceitas pelo sistema, neste caso ignore o passo em questão.
2. Alguns antivírus como o Norton Security, podem reconhecer o Monitor do Driver MFE como um vírus. Neste caso poderá ser solicitada uma ação ao usuário para permitir que o Monitor execute.



3.2 Sistema Operacional Linux

A seguir são descritos os passos para a instalação do Driver MFE para o Sistema Operacional Linux:

1. Acessar o terminal do Linux como root;
2. Descompactar o arquivo instalador-ce-sefaz-driver-linux-32.tar.gz (ou instalador-ce-sefaz-driver-linux-64.tar.gz) executando o comando `tar -vzxf nome do arquivo`;
3. Dentro da pasta, execute o script `instala_driver.sh` como mostra a Figura 5.

```
root@debian:/home/andreissouza/sat_fiscal/driver/driver-mfe-ce-sefaz/instalador-ce-sefaz-driver-l
inux-64# ./instala_driver.sh
Instalação Driver MFE 01.03.45
Instalação Monitor MFE 01.03.01.021
Instalação Comunicador MFE 01.03.01.038
Instalação Biblioteca MFE 01.03.01.029
criando diretorios
.

copiando arquivos

configurando seriais
configurando serviço de comunicacao
habilitando servico
Created symlink from /etc/systemd/system/multi-user.target.wants/cco-ser.service to /lib/systemd/
system/cco-ser.service.
iniciando servico
status do servico
● cco-ser.service - Comunicador - Componente de Comunicacao
   Loaded: loaded (/lib/systemd/system/cco-ser.service; enabled)
   Active: active (running) since Fri 2016-08-05 14:45:43 EDT; 280ms ago
 Main PID: 15510 (runcco-ser.sh)
   CGroup: /system.slice/cco-ser.service
           └─15510 /bin/sh - /opt/sefaz/cco/bin/runcco-ser.sh
             └─15513 ./Comunicador

Aug 05 14:45:43 debian runcco-ser.sh[15510]: INICIANDO Comunicador
Aug 05 14:45:43 debian runcco-ser.sh[15510]: LD_LIBRARY_PATH=
Aug 05 14:45:43 debian runcco-ser.sh[15510]: PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:...bin
Aug 05 14:45:44 debian runcco-ser.sh[15510]: [LogDebug] Instalando captura de logs do Qt.
Hint: Some lines were ellipsized, use -l to show in full.
É NECESSÁRIO REINICIAR O SISTEMA PARA AS CONFIGURAÇÕES FAZEREM EFEITO
```

Figura 5 - Resultado da instalação com sucesso

Os arquivos foram instalados na pasta `/opt/elgin`.



Os componentes da instalação são:

Tabela 1 – Arquivos instalados do Driver MFE

Arquivo	Função
Comunicador	Aplicativo do Comunicador
Monitor	Aplicativo Monitor
libs*.so.*	Bibliotecas auxiliares do Driver MFE
libdllsat.so.*	Biblioteca do SAT
instala_driver.sh	Script de instalação
remove_driver.sh	Script de remoção
cco-ser.ini	Configuração do cco-ser
cco-ser.service	Descriptor do serviço
80-tty-elgin.rules	Configuração das seriais
cco-mon.desktop	Arquivo para startup

4 COMUNICADOR

O Comunicador do Driver MFE é uma aplicação executada como um serviço é responsável por realizar a comunicação com o Módulo MFE através da porta serial emulada sobre uma porta USB.

Sem este serviço em execução a comunicação com o Módulo Fiscal não será possível.

Ele desempenha as seguintes funções:

- Identifica automaticamente quando um MFE é conectado ou desconectado do PDV e coleta informações a seu respeito
- Recebe as funções chamadas via Biblioteca de Funções e realiza a comunicação com o Componente de Segurança embarcado no MFE
- Atualiza as informações exibidas pelo Monitor

5 MONITOR

O Monitor do Driver MFE é uma aplicação que informa, através de ícones na barra de notificações, a disponibilidade ou não de um MFE e permite a visualização de algumas informações.

Ele é responsável por:

- Exibir as informações sobre o MFE coletadas pelo serviço Comunicador MFE
- Exibir as versões de software do DRIVER MFE e seus componentes que estão instaladas.
- Notificar a existência de nova versão de DRIVER MFE e o caminho para download.



Figura 5 – Monitor do Driver MFE na Barra de Notificações



5.1 Ícones

O Monitor do Driver MFE usa os seguintes ícones para representar o MFE:

Tabela 2 - Ícones do Monitor do Driver MFE

	MFE disponível
	MFE não encontrado

5.2 Tela de Status

Ao se clicar no ícone do Monitor é aberta uma tela de status conforme ilustrado abaixo.

Ao se clicar novamente a tela é fechada.



Figura 6 –Tela de status do Monitor do Driver MFE com Módulo Fiscal encontrado



Figura 7 - Tela de status do Monitor do Módulo Fiscal não encontrado

Nesta tela:

- É exibida informação se há MFE disponível ou não
- Ao se clicar no **logo** da Secretaria da Fazenda será aberto o navegador com o endereço **<http://sefaz.ce.gov.br>**
- Ao se clicar em “**Exibir mais informações**” e tela será expandida e serão exibidas outras informações conforme ilustrado abaixo:



Figura 8 - Tela do Monitor do Driver MFE – opção Exibir mais informações

Nesta tela:

- São exibidas informações adicionais
- Ao se clicar em “**Esconder informações**” e tela será recolhida e voltará à forma anterior

As informações exibidas na tela expandida do Monitor MFE são descritas abaixo.

Driver MFE	Versão do Driver MFE
Biblioteca de Funções	Versão da Biblioteca de Funções
Comunicador	Versão do serviço Comunicador
Monitor	Versão do Monitor
Nível GSM	Nível do sinal da rede celular
Wifi Nível	Nível do sinal WIFI
Wifi SSID	Nome da rede WIFI em uso
Cupons pendentes	Quantidade de cupons ainda não enviados para a SEFAZ e que estão armazenados no módulo MFE

Tabela 2 - Informações adicionais do Monitor do Driver MFE



6 BIBLIOTECA DE FUNÇÕES

A Biblioteca de Funções do Driver MFE disponibiliza todas as funções previstas no Projeto CFe, além de funções específicas para o Projeto MFE.

Este documento descreve apenas como devem ser chamadas (protótipos) as Funções CFe e MFE disponíveis na Biblioteca de Funções.

As descrições das funções, seus parâmetros e retornos estão nos documentos de Especificação de Requisitos dos respectivos Projetos:

Projeto CFe (Especificação Nacional)

<http://www.sefaz.ce.gov.br/content/aplicacao/internet/download/gerados/especnacional.asp>

Projeto MFE (Especificação Ceará)

<http://www.sefaz.ce.gov.br/content/aplicacao/internet/download/gerados/especceara.asp>

6.1 Funções do Projeto CFe

6.1.1 Função AtivarSAT

Usada no processo de ativação do MFE e serve para informar:

- Qual será tipo de certificado usado na ativação: AC-SAT ou ICP-Brasil
- Código de Ativação
- CNPJ do Contribuinte
- Unidade Federativa

Protótipo da função:

```
char* AtivarSAT(  
    int iNumSessao,  
    int iSubCmd,  
    char* pcCodAtivacao,  
    char* pcCnpj,  
    int iUF  
)
```



Parâmetros:

iNumSessao: Número da sessão.
iSubCmd: Tipo de Certificado: AC-SAT ou ICP-BRASIL
pcCodAtivacao: Senha definida pelo contribuinte.
pcCnpj: CNPJ do contribuinte, somente números.
iUF: Código do Estado. Ceará é 23.

6.1.2 Função ComunicarCertificadoICPBrasil

Usada no processo de ativação do MFE, quando escolhido certificado tipo ICP-BRASIL, ou na renovação de um certificado digital ICP-BRASIL.

Protótipo da função:

```
char* ComunicarCertificadoICPBRASIL(  
    int iNumSessao,  
    char* pcCodAtivacao,  
    char* pcCertificado  
)
```

Parâmetros:

iNumSessao: Número da sessão.
pcCodAtivacao: Senha definida pelo contribuinte.
pcCertificado: Certificado Digital criado pela Autoridade Certificadora ICP-Brasil.

6.1.3 Função EnviarDadosVenda

Envia os dados de uma venda para que o MFE gere um CFe de venda.

Protótipo da função:

```
char* EnviarDadosVenda(  
    int iNumSessao,  
    char* pcCodAtivacao,  
    char* pcDadosVenda  
)
```



Parâmetros:

iNumSessao: Número da sessão.
pcCodAtivacao: Senha definida pelo contribuinte.
pcDadosVenda: XML com os dados de venda.

6.1.4 Função CancelarUltimaVenda

Envia os dados de um cancelamento de venda para que o MFE gere um CFe de cancelamento de venda.

Protótipo da função:

```
char* CancelarUltimaVenda (  
    int iNumSessao,  
    char* pcCodAtivacao,  
    char* pcChaveCfe,  
    char* pcDadosCanc  
)
```

Parâmetros:

iNumSessao: Número da sessão.
pcCodAtivacao: Senha definida pelo contribuinte.
pcChaveCfe: Chave de acesso do CFe a ser cancelado.
pcDadosCanc: XML com os dados de cancelamento.

6.1.5 Função ConsultarSAT

Função usada para verificar se há comunicação com o MFE.

Protótipo da função:

```
char* ConsultarSAT (  
    int iNumSessao  
)
```

Parâmetros:

iNumSessao: Número da sessão.



6.1.6 Função TesteFimAFim

Função usada para realizar a emissão de um CFe de teste e verificar se o processo de emissão de CFes está funcionando.

Protótipo da função:

```
char* TesteFimAFim (  
    int iNumSessao,  
    char* pcCodAtivacao,  
    char* pcDadosVenda  
)
```

Parâmetros:

iNumSessao: Número da sessão.
pcCodAtivacao: Senha definida pelo contribuinte.
pcDadosVenda: XML com os dados da venda de teste.

6.1.7 Função ConsultarStatusOperacional

Função usada para obter informações sobre o estado de operação do MFE.

Protótipo da função:

```
char* ConsultarStatusOperacional (  
    int iNumSessao,  
    char* pcCodAtivacao  
)
```

Parâmetros:

iNumSessao: Número da sessão.
pcCodAtivacao: Senha definida pelo contribuinte.



6.1.8 Função ConsultarNumeroSessao

Função usada para obter novamente os dados retornados na última sessão.

Protótipo da função:

```
char* ConsultarNumeroSessao (  
    int iNumSessao,  
    char* pcCodAtivacao,  
    int luiNumSessaoConsultada  
)
```

Parâmetros:

iNumSessao:	Número da sessão.
pcCodAtivacao:	Senha definida pelo contribuinte.
iNumSessaoConsultada:	Número da sessão a ser consultada.

6.1.9 Função ConfigurarInterfaceDeRede

Função usada para configurar a comunicação do MFE com a internet.

Protótipo da função:

```
char* ConfigurarInterfaceDeRede (  
    int iNumSessao,  
    char* pcCodAtivacao,  
    char* pcDadosConfiguracao  
)
```

Parâmetros:

iNumSessao:	Número da sessão.
pcCodAtivacao:	Senha definida pelo contribuinte.
pcDadosConfiguracao:	XML com os dados de configuração das interfaces do MFE.



6.1.10 Função AssociarAssinatura

Função que vincula a assinatura do Aplicativo Comercial com o Contribuinte.

Usada na ativação do MFE ou quando for trocado o fornecedor do Aplicativo Comercial.

Protótipo da função:

```
char* AssociarAssinatura (  
    int iNumSessao,  
    char* pcCodAtivacao,  
    char* pcCnpj,  
    char* lpcAssinaturaCnpps  
)
```

Parâmetros:

iNumSessao:	Número da sessão.
pcCodAtivacao:	Senha definida pelo contribuinte.
pcCnpj:	CNPJ da empresa desenvolvedora do Aplicativo Comercial + CNPJ do Contribuinte.
pcAssinaturaCnpps:	Assinatura digital conjunto “CNPJ Software House” + “CNPJ do Contribuinte”.

6.1.11 Função AtualizarSoftwareSAT

Função para atualizar o Software Básico do MFE.

Protótipo da função:

```
char* AtualizarSoftwareSAT (  
    int iNumSessao,  
    char* pcCodAtivacao  
)
```

Parâmetros:

iNumSessao:	Número da sessão.
pcCodAtivacao:	Senha definida pelo contribuinte.



6.1.12 Função ExtrairLogs

Função para obter o log de operação do MFE.

Protótipo da função:

```
char* ExtrairLogs (  
    int iNumSessao,  
    char* pcCodAtivacao  
)
```

Parâmetros:

iNumSessao: Número da sessão.
pcCodAtivacao: Senha definida pelo contribuinte.

6.1.13 Função BloquearSAT

Função para bloquear as funções fiscais do MFE.

Protótipo da função:

```
char* BloquearSAT (  
    int iNumSessao,  
    char* pcCodAtivacao  
)
```

Parâmetros:

iNumSessao: Número da sessão.
pcCodAtivacao: Senha definida pelo contribuinte.



6.1.14 Função DesbloquearSAT

Função para desbloquear as funções fiscais do MFE.

Protótipo da função:

```
char* DesbloquearSAT (  
    int iNumSessao,  
    char* pcCodAtivacao  
)
```

Parâmetros:

iNumSessao: Número da sessão.
pcCodAtivacao: Senha definida pelo contribuinte.

6.1.15 Função TrocarCodigoDeAtivacao

Função para trocar o código de ativação escolhido pelo Contribuinte na ativação do MFE.

Protótipo da função:

```
char* TrocarCodigoDeAtivacao (  
    int iNumSessao,  
    char* pcCodAtivacao,  
    int iOpcao,  
    char* pcNovoCod,  
    char* pcConfirmaNovoCod  
)
```

Parâmetros:

iNumSessao: Número da sessão.
pcCodAtivacao: Senha definida pelo contribuinte.
iOpcao: Informa se está sendo usado o código de emergência ou não.
pcNovoCod: Novo código de ativação.
pcConfirmaNovoCod: Confirmação do novo código de ativação.



6.2 Funções do Projeto MFE

6.2.1 Função ConfigurarInterfaceDeRedeMFE

Função para configurar as interfaces de rede do MFE.

Permite a configuração de várias interfaces simultaneamente incluindo a de Modem Celular (não existente no projeto CFe).

Protótipo da função:

```
char* ConfigurarInterfaceDeRedeMFE (  
    int iNumSessao,  
    char* pcCodAtivacao,  
    char* pcDadosConfiguracao  
)
```

Parâmetros:

iNumSessao: Número da sessão.

pcCodAtivacao: Senha definida pelo contribuinte.

pcDadosConfiguracao: XML com os dados de configuração das interfaces.

6.2.2 Função ConsultarStatusOperacionalMFE

Função usada para obter informações sobre o estado de operação do MFE, incluindo informações sobre o hardware adicional não existente no Projeto CFe.

Protótipo da função:

```
char* ConsultarStatusOperacionalMFE (  
    int iNumSessao,  
    char* pcCodAtivacao  
)
```

Parâmetros:

iNumSessao: Número da sessão.

pcCodAtivacao: Senha definida pelo contribuinte.

pcDadosConfiguracao: XML com os dados de configuração das interfaces.



6.2.3 Função EnviarStatusAC

A função EnviarStatusAC é utilizada pelo AC para realizar a indicação do estado operacional do mesmo para o sistema MFE através do envio de informações de versão e fabricante. Essa função é enviada pelo DRS e consumida pelo Componente de Comunicação (CCO).

Protótipo da função:

```
void EnviarStatusAC(
```

```
    iNumSessao,
```

```
    char* pcCodAtivacao,
```

```
    char* pcDescFornecedor,
```

```
    char* pcVersaoAC
```

```
);
```

Parâmetros:

iNumSessao:	Número da sessão.
pcCodAtivacao:	Senha definida pelo contribuinte.
pcDescFornecedor:	Nome do fornecedor do AC.
pcVersaoAC:	Versão do AC.



7 PROTÓTIPOS DAS FUNÇÕES

Abaixo segue a lista dos protótipos das funções dos Projetos CFE e MFE:

```
char* _STDCALL AtivarSAT(int iNumSessao, int iSubCmd, const char* pcCodAtivacao,
const char* pcCnpj, int iUF );
char* _STDCALL ComunicarCertificadoICPBRASIL(int iNumSessao, const char*
pcCodAtivacao, const char* pcCertificado );
char* _STDCALL EnviarDadosVenda(int iNumSessao, const char* pcCodAtivacao, const
char* pcDadosVenda );
char* _STDCALL CancelarUltimaVenda(int iNumSessao, const char* pcCodAtivacao,
const char* pcChaveCfe, const char* pcDadosCanc );
char* _STDCALL ConsultarSAT(int iNumSessao );
char* _STDCALL TesteFimAFim(int iNumSessao, const char*
pcCodAtivacao, const char* pcDadosVenda );
char* _STDCALL ConsultarStatusOperacional(int iNumSessao, const char*
pcCodAtivacao );
char* _STDCALL ConsultarNumeroSessao (int iNumSessao, const char* pcCodAtivacao,
int iNumSessaoConsultada );
char* _STDCALL ConfigurarInterfaceDeRede (int iNumSessao, const char*
pcCodAtivacao, const char* pcDadosConfiguracao );
char* _STDCALL AssociarAssinatura (int iNumSessao, const char* pcCodAtivacao,
const char* pcCnpj, const char* pcAssinaturaCnpj );
char* _STDCALL AtualizarSoftwareSAT(int iNumSessao, const char* pcCodAtivacao );
char* _STDCALL ExtrairLogs(int iNumSessao, const char* pcCodAtivacao );
char* _STDCALL BloquearSAT (int iNumSessao, const char* pcCodAtivacao );
char* _STDCALL DesbloquearSAT (int iNumSessao, const char* pcCodAtivacao );
char* _STDCALL TrocarCodigoDeAtivacao(int iNumSessao, const char* pcCodAtivacao,
int iOpcao, const char* pcNovoCod, const char* pcConfirmaNovoCod );
char* _STDCALL ConsultarStatusOperacionalMFE(int iNumSessao, const char*
pcCodAtivacao );
char* _STDCALL ConfigurarInterfaceDeRedeMFE (int iNumSessao, const char*
pcCodAtivacao, const char* pcDadosConfiguracao );
char* _STDCALL EnviarStatusAC (int iNumSessao, const char* pcCodAtivacao, const
char* pcFabricante, const char * pcVersaoAC);
```