

# MANUAL DO DESENVOLVEDOR

EASYS@T



# 1 Índice

---

1	Índice.....	2
2	API SAT.dll .....	3
2.1.1	Ativar SAT.....	3
2.1.2	ComunicarCertificadoICPBRASIL – Envio do Certificado Criado pela ICPBRASIL.....	4
2.1.3	EnviarDadosVenda – Emissão da venda.....	5
2.1.4	CancelarUltimaVenda – Cancelamento dA venda.....	6
2.1.5	ConsultarSAT – Consulta entre AC e Equipamento SAT .....	8
2.1.6	TesteFimAFim – Teste fim-a-fim (Teste de envio de dados de venda)8	
2.1.7	ConsultarStatusOperacional – Consulta do Status Operacional do Equipamento SAT .....	9
2.1.8	ConsultarNumeroSessao – Comando de Consulta do número de sessão	10
2.1.9	ConfigurarInterfaceDeRede – Configuração da interface de comunicação do Equipamento SAT .....	11
2.1.10	AssociarAssinatura – Associação da Assinatura do Aplicativo Comercial.....	12
2.1.11	AtualizarSoftwareSAT – Atualização do software do Equipamento SAT	13
2.1.12	ExtrairLogs – Extração de logs do SAT .....	14
2.1.13	BloquearSAT .....	15
2.1.14	DesbloquearSAT .....	16
2.1.15	TrocarCodigoDeAtivacao .....	16
3	Referências .....	18

## 2 API SAT.dll

---

A interface de comunicação entre o Aplicativo Cliente e o [EASYS@T](#) é feito pela dll SAT.dll instalada em C:\SAT. Apresentamos a seguir as funções disponíveis na API para serem chamadas.

Para que o Aplicativo Comercial consiga utilizar as funções é preciso carregar a dll dentro do código gerado. O modo como isso é feito fica a encargo do desenvolvedor e é específico da linguagem de programação utilizada.

As funções estão separadas pela funcionalidade.

Grande parte dos textos descrevendo cada funcionalidade foi retirado da Especificação apresentada na Referência 1. Eles foram transcritos aqui com o intuito de facilitar o acesso às informações. Alguns tópicos fazem referência à itens da Referência 1, para maiores detalhes é preciso consultá-los.

### 2.1.1 Ativar SAT

Esta função faz parte do processo de ativação do Equipamento SAT e será responsável por enviar ao SAT qual o tipo de ativação será efetuada pelo Contribuinte.

Para a funcionalidade de Ativar o SAT deve-se usar a função:

```
String AtivarSAT (long numeroSessao, long subComando, String  
codigoDeAtivacao, String cnpj, long cUF);
```

#### 2.1.1.1 Parâmetros

Os parâmetros passados são:

- » **numeroSessao[type long]**: Número aleatório gerado pelo AC para controle da comunicação;
- » **subComando[type long]**: Identificador do tipo de Certificado;
- » **codigoDeAtivacao[type String]**: Senha definida pelo contribuinte no software de ativação. Deve conter de 8 a 32 caracteres alfanuméricos;
- » **cnpj[type String]**: CNPJ do contribuinte, somente números;
- » **cUF[type long]**: Código do Estado da Federação onde o SAT será ativado.

#### 2.1.1.2 Retorno

A função deverá retornar um valor do tipo string contendo os dados separados por pipe “|”. O valor retornado poderá ser diferente dependendo do valor informado no parâmetro “subComando”. Segue a estrutura do retorno de dados:

**Retorno:** “numeroSessao|EEEEEE|mensagem|cod|mensagemSEFAZ|CSR”, onde:

- » **numeroSessao:** Número aleatório gerado pelo AC para controle da comunicação;
- » **EEEEEE:** O código EEEEE é substituído por um EEEEE dentre os possíveis códigos de retorno descritos no item 6.2 da Especificação (Referência 1);
- » **mensagem:** Descrições dos códigos EEEEE;
- » **cod:** Código de referência de cada mensagemSEFAZ;
- » **mensagemSEFAZ:** Mensagem de texto enviada pela mensagemSEFAZ referente ao “Envio de avisos ao usuário”;
- » **CSR:** CSR arquivo (codificado em base 64) contendo as informações para a solicitação do Certificado Digital.

Em caso de erro na função, deverá ser retornado apenas os cinco primeiros campos da tabela acima (tabela de descrição dos campos do retorno), o retorno ficará da seguinte forma:

**Retorno:** “numeroSessao|EEEEEE|mensagem|cod|mensagemSEFAZ”

### 2.1.2 ComunicarCertificadoICPBRASIL – Envio do Certificado Criado pela ICPBRASIL

Esta função faz parte do processo de ativação do Equipamento SAT e será responsável por enviar ao SAT o certificado recebido da Autoridade Certificadora ICP-Brasil.

Para essa funcionalidade deve-se utilizar a função:

```
String ComunicarCertificadoICPBRASIL (long numeroSessao,
String codigoDeAtivacao, String certificado)
```

#### 2.1.2.1 Parâmetros

Os parâmetros passados são:

- » **numeroSessao[type long]:** Número aleatório gerado pelo AC para controle da comunicação;
- » **codigoDeAtivacao[type String]:** Senha definida pelo contribuinte no software de ativação;
- » **certificado:[type String]** Certificado Digital criado pela Autoridade Certificadora – ICP-Brasil.

#### 2.1.2.2 Retorno

função deverá retornar um valor do tipo string contendo os dados separados por pipe “|”. O valor retornado poderá ser diferente dependendo do valor informado no parâmetro “subComando”. Segue a estrutura do retorno de dados:

**Retorno:** “numeroSessao|EEEE|mensagem|cod|mensagemSEFAZ|CSR”, onde:

- » **numeroSessao:** Número aleatório gerado pelo AC para controle da comunicação;
- » **EEEE:** O código EEEE é substituído por um EEEE dentre os possíveis códigos de retorno descritos no item 6.2 da Especificação (Referência 1);
- » **mensagem:** Descrições dos códigos EEEE;
- » **cod:** Código de referência de cada mensagemSEFAZ;
- » **mensagemSEFAZ:** Mensagem de texto enviada pela mensagemSEFAZ referente ao “Envio de avisos ao usuário”.

### 2.1.3 EnviarDadosVenda – Emissão da venda

Esta função faz parte do processo de envio dos dados de venda do AC para o Equipamento SAT.

Para essa funcionalidade deve-se utilizar a função:

```
String EnviarDadosVenda (long numeroSessao, String  
codigoDeAtivacao, String dadosVenda);
```

#### 2.1.3.1 Parâmetros

Um dos parâmetros é a *mensagem* e ele possui um formato específico. Os campos obrigatórios, bem como os tamanhos e formatos de cada um, devem seguir a especificação do leiaute do arquivo de venda (descrita no Capítulo 4 da Referência 1). O AC deverá enviar os dados em formato XML com suas respectivas tags.

Os parâmetros passados são:

- » **numeroSessao[type long]:** Número aleatório gerado pelo AC para controle da comunicação;
- » **codigoDeAtivacao[type String]:** Senha definida pelo contribuinte no software de ativação;
- » **dadosVenda[type String]:** Refere-se aos dados de venda gerados pelo AC.

#### 2.1.3.2 Retorno

A função deverá retornar um valor do tipo string contendo os dados separados por pipe “|”. Segue a estrutura do retorno de dados. Segue a estrutura de retorno com sucesso:

**Retorno:**

“numeroSessao|EEEE|CCCC|mensagem|cod|mensagemSEFAZ|arquivoCFeBase64|timeStamp|chaveConsulta|valorTotalCFe|CPFCNPJValue|assinaturaQRCode”,  
onde:

- » **numeroSessao:** Número aleatório gerado pelo AC para controle da comunicação;
- » **EEEEEE:** O código EEEEE é substituído por um EEEEE dentro os possíveis códigos de retorno descritos no item 6.2 da Especificação;
- » **CCCC:** Em caso de alerta (EEEEEE = 06000) ou rejeição nos dados de venda (EEEEEE = 06010) o código CCCC será substituído por um dentre os possíveis códigos de retorno (item 6.3.1 da Referência 1). Nos demais casos este campo retornará o código "0000";
- » **mensagem:** São as descrições dos códigos EEEEE;
- » **cod:** Código de referência de cada mensagemSEFAZ;
- » **mensagemSEFAZ:** Mensagem de texto enviada pela *Sefaz* referente ao "Envio de avisos ao usuário";
- » **ArquivoCFeBase64:** arquivo em XML assinado pelo SAT em formato idêntico ao que o SAT enviará para a *Sefaz*. Codificado em Base 64;
- » **timeStamp:** Data e hora da emissão no formato: AAAAMMDDHHMMSS;
- » **chaveConsulta:** Chave de acesso do CF-e-SAT precedida do literal "CFe";
- » **valorTotalCFe:** Valor total do CF-e calculado pelo SAT. Campo presente no leiaute do arquivo de venda;
- » **CPFCNPJValue:** Número do CPF ou CNPJ do adquirente no seguinte sem pontos, traços e barras;
- » **assinaturaQRCODE:** Conteúdo do Campo "assinaturaQRCODE" presente no leiaute do arquivo de venda.

Em caso de erro na função, deverão ser retornados apenas os seis primeiros campos da tabela acima (tabela de descrição dos campos do retorno). O retorno ficará da seguinte forma:

**Retorno:** "numeroSessao|EEEEEE|CCCC|mensagem|cod|mensagemSEFAZ"

#### 2.1.4 CancelarUltimaVenda – Cancelamento dA venda

O envio dos dados de cancelamento da venda ocorrerá de acordo com as definições a seguir.

Para essa funcionalidade deve-se utilizar a função:

```
String CancelarUltimaVenda (long numeroSessao, String
codigoDeAtivacao, String chave, String dadosCancelamento);
```

##### 2.1.4.1 Parâmetros

O envio dos dados de cancelamento do AC para o Equipamento SAT deverá ser através da função CancelarUltimaVenda, e segue a seguinte estrutura:

- » **numeroSessao[type long]:** Número aleatório gerado pelo AC para controle da comunicação;
- » **codigoDeAtivacao[type String]:** Senha definida pelo contribuinte no software de ativação;
- » **chave[type String]:** Chave de acesso do CF-e-SAT a ser cancelado precedida do literal "CFe";
- » **dadosCancelamento[type String]:** XML de cancelamento gerado pela AC e com dados para compor o CF-e-SAT de cancelamento.

#### 2.1.4.2 Retorno

A função deverá retornar um valor do tipo string contendo os dados separados por pipe "|". Segue o seguinte formato:

##### Retorno:

"numeroSessao|EEEE|CCCC|mensagem|cod|mensagemSEFAZ|arquivoCFeBase64|timeStamp|chaveConsulta|valorTotalCFe|CPFCNPJValue|assinaturaQRCode",  
onde:

- » **numeroSessao:** Número aleatório gerado pelo AC para controle da comunicação;
- » **EEEE:** O código EEEE é substituído por um EEEE dentre os possíveis códigos de retorno descritos no item 6.2 da Especificação;
- » **CCCC:** Em caso de rejeição nos dados de cancelamento (EEEE = 07007) o código CCCC será substituído por um CCCC dentre os possíveis códigos de retorno (item 6.3.2 da Referência 1). Nos demais casos este campo retornará o código "0000";
- » **mensagem:** São as descrições dos códigos EEEE;
- » **cod:** Código de referência de cada mensagemSEFAZ;
- » **mensagemSEFAZ:** Mensagem de texto enviada pela *Sefaz* referente ao "Envio de avisos ao usuário";
- » **ArquivoCFeBase64:** Arquivo em XML assinado pelo SAT em formato idêntico ao que o SAT enviará para a *Sefaz*. Codificado em Base 64;
- » **timeStamp:** Data e hora da emissão no formato: AAAAMMDDHHMMSS;
- » **chaveConsulta:** Chave de acesso do CF-e-SAT precedida do literal "CFe";
- » **valorTotalCFe:** Valor total do CF-e calculado pelo SAT. Campo presente no leiaute do arquivo de venda;
- » **CPFCNPJValue:** Número do CPF ou CNPJ do adquirente no seguinte sem pontos, traços e barras;



- » **assinaturaQRCODE:** Conteúdo do Campo "assinaturaQRCODE" presente no leiaute do arquivo de cancelamento.

Em caso de erro na função, deverá ser retornado apenas os cinco primeiros campos descritos acima, o retorno ficará da seguinte forma:

**Retorno:** "numeroSessao|EEEE|CCCC|mensagem|cod|mensagemSEFAZ"

#### 2.1.5 ConsultarSAT – Consulta entre AC e Equipamento SAT

Esta função é usada para testes de comunicação entre o AC e o Equipamento SAT.

Para essa funcionalidade deve-se utilizar a função:

```
String ConsultarSAT (long numeroSessao);
```

##### 2.1.5.1 Parâmetros

Os parâmetros são:

- » **numeroSessao[type long]:** Número aleatório gerado pelo AC para controle da comunicação;

##### 2.1.5.2 Retorno

A função deverá retornar um valor do tipo string contendo os dados separados por pipe "|". Segue a estrutura do retorno de dados:

**Retorno:** "numeroSessao|EEEE|mensagem|cod|mensagemSEFAZ", onde:

- » **numeroSessao:** Número aleatório gerado pelo AC para controle da comunicação;
- » **EEEE:** O código EEEE é substituído por um EEEE dentre os possíveis códigos de retorno descritos no item 6.2 da Especificação (Referência 1);
- » **mensagem:** São as descrições dos códigos EEEE;
- » **cod:** Código de referência de cada mensagemSEFAZ;
- » **mensagemSEFAZ:** Mensagem de texto enviada pela *Sefaz* referente ao "Envio de avisos ao usuário".

#### 2.1.6 TesteFimAFim – Teste fim-a-fim (Teste de envio de dados de venda)

Para essa funcionalidade deve-se utilizar a função:

```
String TesteFimAFim (long numeroSessao, String  
codigoDeAtivacao, String dadosVenda);
```



### 2.1.6.1 Parâmetros

Os parâmetros são:

- » **numeroSessao[type long]:** Número aleatório gerado pelo AC para controle da comunicação;
- » **codigoDeAtivacao[type String]:** Senha definida pelo contribuinte no software de ativação;
- » **dadosVenda[typr String]:** Refere-se aos dados de venda fictícios gerados pelo AC.

### 2.1.6.2 Retorno

A função deverá retornar um valor do tipo string contendo os dados separados por pipe "|". Segue a estrutura do retorno de dados com sucesso:

**Retorno:**

“numeroSessao|EEEE|mensagem|cod|mensagemSEFAZ|arquivoCFeBase64|timeStamp|num-DocFiscal|chaveConsulta”, onde:

- » **numeroSessao:** Número aleatório gerado pelo AC para controle da comunicação;
- » **EEEE:** O código EEEE é substituído por um EEEE dentre os possíveis códigos de retorno descritos no item 6.2 da Especificação (Referência 1);
- » **mensagem:** São as descrições dos códigos EEEE;
- » **cod:** Código de referência de cada mensagemSEFAZ;
- » **mensagemSEFAZ:** Mensagem de texto enviada pela *Sefaz* referente ao "Envio de avisos ao usuário";
- » **ArquivoCFeBase64:** arquivo em XML assinado pelo SAT em formato idêntico ao que o SAT enviará para a *Sefaz*. Codificado em Base 64;
- » **timeStamp:** Data e hora da emissão no formato: AAAAMMDDHHMMSS;
- » **numDocFiscal:** Número do Cupom Fiscal Eletrônico - SAT. Campo "nCFe" no leiaute do arquivo de venda;
- » **chaveConsulta:** Chave de acesso do CF-e-SAT precedida do literal "CFe".

Em caso de erro na função, deverá ser retornado apenas os cinco primeiros campos da tabela acima. O retorno ficará da seguinte forma:

**Retorno:** “numeroSessao|EEEE|mensagem|cod|mensagemSEFAZ”

### 2.1.7 ConsultarStatusOperacional – Consulta do Status Operacional do Equipamento SAT

Para essa funcionalidade deve-se utilizar a função:

```
String ConsultarStatusOperacional (long numeroSessao, String codigoDeAtivacao);
```

#### 2.1.7.1 Parâmetros

Os parâmetros são:

- » **numeroSessao[type long]:** Número aleatório gerado pelo AC para controle da comunicação;
- » **codigoDeAtivacao[type String]:** Senha definida pelo contribuinte no software de ativação.

#### 2.1.7.2 Retorno

A função deverá retornar um valor do tipo string contendo os dados separados por pipe "|". Segue a estrutura do retorno de dados:

**Retorno:**

“numeroSessao|EEEE|mensagem|cod|mensagemSEFAZ|ConteudoRetorno”, onde:

- » **numeroSessao:** Número aleatório gerado pelo AC para controle da comunicação;
- » **EEEE:** O código EEEE é substituído por um EEEE dentre os possíveis códigos de retorno descritos no item 6.2 da Especificação (Referência 1);
- » **mensagem:** São as descrições dos códigos EEEE;
- » **cod:** código de referência de cada mensagemSEFAZ;
- » **mensagemSEFAZ:** Mensagem de texto enviada pela *Sefaz* referente ao "Envio de avisos ao usuário";
- » **ConteudoRetorno:** O “Conteúdo de retorno” consiste nas informações sobre o status do SAT.

Para maiores informações sobre o campo ConteudoRetorno e sua composição consulte a Referência 1, no item 6.1.7.2 a Tabela de Informações do status do SAT.

Em caso de erro na função, deverá ser retornado apenas os cinco primeiros campos da tabela acima.

#### 2.1.8 ConsultarNumeroSessao – Comando de Consulta do número de sessão

Utilizada para a AC consultar se a ultima requisição foi processada, caso o retorno não tenha sido recebido. O SAT retonara o resultado da ultima sessão consultada.

Para essa funcionalidade deve-se utilizar a função:

```
String ConsultarNumeroSessao (long numeroSessao, String codigoDeAtivacao, long cNumeroDeSessao);
```

### 2.1.8.1 Parâmetros

Os parâmetros são:

- » **numeroSessao[type long]:** Número aleatório gerado pelo AC para controle da comunicação;
- » **codigoDeAtivacao[type String]:** Senha definida pelo contribuinte no software de ativação;
- » **cNumeroDeSessao[type long]:** Número de sessão a ser consultado no SAT-CF-e.

### 2.1.8.2 Retorno

Caso a função já tenha sido processada o retorno será exatamente igual ao retorno da função processada.

Caso a sessão não seja reconhecida pelo SAT o retorno será da seguinte forma:

**Retorno:** “numeroSessao|EEEEEE|mensagem|cod|mensagemSEFAZ”, sendo:

- » **numeroSessao:** Número aleatório gerado pelo AC para controle da comunicação;
- » **EEEEEE:** O código EEEEE é substituído por um EEEEE dentre os possíveis códigos de retorno descritos no item 6.2 da Especificação (Referência 1);
- » **mensagem:** São as descrições dos códigos EEEEE;
- » **cod:** Código de referência de cada mensagemSEFAZ;
- » **mensagemSEFAZ:** Mensagem de texto enviada pela *Sefaz* referente ao "Envio de avisos ao usuário";
- » **ConteudoRetorno:** O "Conteúdo de retorno" consiste nas informações sobre o status do SAT.

### 2.1.9 ConfigurarInterfaceDeRede – Configuração da interface de comunicação do Equipamento SAT

O AC, ou outro software fornecido pelo Fabricante, poderá configurar a interface de comunicação do Equipamento SAT com a rede local do estabelecimento comercial através do envio de um arquivo de configuração no padrão XML.

No parâmetro “codigoDeAtivacao” deve ser informado oito vezes o caracter “0” (zero) caso o equipamento SAT-CF-e ainda não tenha sido ativado. Após sua ativação, deve ser informado e conferido o código de ativação.

Para essa funcionalidade deve-se utilizar a função:

```
String Configurar InterfaceDeRede (long numeroSessao, String  
codigoDeAtivacao, String dadosConfiguracao);
```

#### 2.1.9.1 Parâmetros

Os parâmetros são:

- » **numeroSessao[type long]:** Número aleatório gerado pelo AC para controle da comunicação;
- » **codigoDeAtivacao[type String]:** Senha definida pelo contribuinte no software de ativação;
- » **dadosConfiguracao[type String]:** Arquivo de configuração no formato XML.

A forma de construir o XML de configuração é apresentada na Referência 1, no item 6.1.9.1, na Tabela 34 – Dados da Configuração (Descrição dos parâmetros do Arquivo de configuração de Rede).

#### 2.1.9.2 Retorno

A função deverá retornar um valor do tipo string contendo os dados separados por pipe "|".

Segue a estrutura do retorno de dados:

**Retorno:** "numeroSessao|EEEE|mensagem|cod|mensagemSEFAZ", onde:

- » **numeroSessao:** Número aleatório gerado pelo AC para controle da comunicação;
- » **EEEE:** O código EEEE é substituído por um EEEE dentre os possíveis códigos de retorno descritos no item 6.2 da Especificação (Referência 1);
- » **mensagem:** São as descrições dos códigos EEEE;
- » **cod:** Código de referência de cada mensagemSEFAZ;
- » **mensagemSEFAZ:** Mensagem de texto enviada pela *Sefaz* referente ao "Envio de avisos ao usuário".

#### 2.1.10 AssociarAssinatura – Associação da Assinatura do Aplicativo Comercial

O contribuinte deverá associar a assinatura do Aplicativo Comercial com o SAT através da função AssociarAssinatura.

Para essa funcionalidade deve-se utilizar a função:

String AssociarAssinatura (**long** numeroSessao, String  
codigoDeAtivacao, String CNPJvalue, String assinaturaCNPJs);

##### 2.1.10.1 Parâmetros

Os parâmetros são:

- » **numeroSessao[type long]:** Número aleatório gerado pelo AC para controle da comunicação;
- » **codigoDeAtivacao[type String]:** Senha definida pelo contribuinte no software de ativação;
- » **CNPJvalue[type String]:** CNPJ da empresa desenvolvedora do Aplicativo Comercial + CNPJ do Emitente;
- » **assinaturaCNPJs[type String]:** Assinatura digital conjunto “CNPJ Software House” + “CNPJ do estabelecimento comercial”.

#### **2.1.10.2 Retorno**

Redação atual, efeitos até 31.12.15

A resposta de associação da Assinatura Digital do AC só retornará com sucesso após a *Sefaz* enviar uma mensagem de sucesso na operação. A função deverá retornar um valor do tipo string contendo os dados separados por pipe “|”. Segue a estrutura do retorno de dados:

**Retorno:** “numeroSessao|EEEEEE|mensagem|cod|mensagemSEFAZ”, onde:

- » **numeroSessao:** Número aleatório gerado pelo AC para controle da comunicação;
- » **EEEEEE:** O código EEEEE é substituído por um EEEEE dentre os possíveis códigos de retorno descritos no item 6.2 da Especificação (Referência 1);
- » **mensagem:** São as descrições dos códigos EEEEE;
- » **cod:** Código de referência de cada mensagemSEFAZ;
- » **mensagemSEFAZ:** Mensagem de texto enviada pela *Sefaz* referente ao "Envio de avisos ao usuário".

#### **2.1.11 AtualizarSoftwareSAT – Atualização do software do Equipamento SAT**

Utilizada para a atualização do software do SAT. Utiliza-se a seguinte função:

`String AtualizarSoftwareSAT (long numeroSessao, String codigoDeAtivacao);`

##### **2.1.11.1 Parâmetros**

Os parâmetros são:

- » **numeroSessao[type long]:** Número aleatório gerado pelo AC para controle da comunicação;
- » **codigoDeAtivacao[type String]:** Senha definida pelo contribuinte no software de ativação.

#### **2.1.11.2 Retorno**

O Aplicativo Comercial deverá esperar a resposta da Atualização do Equipamento SAT antes de permitir execução de qualquer outra função.

A função deverá retornar um valor do tipo string contendo os dados separados por pipe "|".

Segue a estrutura do retorno de dados:

**Retorno:** "numeroSessao|EEEE|mensagem|cod|mensagemSEFAZ", onde:

- » **numeroSessao:** Número aleatório gerado pelo AC para controle da comunicação;
- » **EEEE:** O código EEEE é substituído por um EEEE dentre os possíveis códigos de retorno descritos no item 6.2 da Especificação (Referência 1);
- » **mensagem:** São as descrições dos códigos EEEE;
- » **cod:** Código de referência de cada mensagemSEFAZ;
- » **mensagemSEFAZ:** Mensagem de texto enviada pela *Sefaz* referente ao "Envio de avisos ao usuário".

#### **2.1.12 ExtrairLogs – Extração de logs do SAT**

Utilizada para extrair os logs de registro do SAT. Para essa funcionalidade utiliza-se a função:

```
String ExtrairLogs (long numeroSessao, String codigoDeAtivacao);
```

##### **2.1.12.1 Parâmetros**

Redação atual, efeitos até 31.12.15

Os parâmetros são:

- » **numeroSessao[type long]:** Número aleatório gerado pelo AC para controle da comunicação;
- » **codigoDeAtivacao[type String]:** Senha definida pelo contribuinte no software de ativação.

##### **2.1.12.2 Retorno**

A função deverá retornar um valor do tipo string contendo os dados separados por pipe "|".

Segue a estrutura do retorno de dados.

**Retorno:** "numeroSessao|EEEE|mensagem|cod|mensagemSEFAZ|Arquivo de log em base64", onde:

- » **numeroSessao:** Número aleatório gerado pelo AC para controle da comunicação;
- » **EEEEEE:** O código EEEEE é substituído por um EEEEE dentre os possíveis códigos de retorno descritos no item 6.2 da Especificação (Referência 1);
- » **mensagem:** São as descrições dos códigos EEEEE;
- » **cod:** Código de referência de cada mensagemSEFAZ;
- » **mensagemSEFAZ:** Mensagem de texto enviada pela *Sefaz* referente ao "Envio de avisos ao usuário";
- » **Arquivo de log em base64:** Arquivo de registros de operações do SAT codificado em base64.

Em caso de erro na função, deverá ser retornado apenas os cinco primeiros campos da tabela acima. O retorno ficará da seguinte forma:

**Retorno:** “numeroSessao|EEEEEE|mensagem|cod|mensagemSEFAZ”

#### 2.1.13 BloquearSAT

O bloqueio operacional do Equipamento SAT será realizado através da função BloquearSAT.

Para essa funcionalidade deve-se utilizar a função:

`String BloquearSAT (long numeroSessao, String codigoDeAtivacao);`

##### 2.1.13.1 Parâmetros

Os parâmetros são:

- » **numeroSessao[type long]:** Número aleatório gerado pelo AC para controle da comunicação;
- » **codigoDeAtivacao[type String]:** Senha definida pelo contribuinte no software de ativação.

##### 2.1.13.2 Retorno

A função deverá retornar um valor do tipo string contendo os dados separados por pipe “|”.

Segue a estrutura do retorno de dados:

**Retorno:** “numeroSessao|EEEEEE|mensagem|cod|mensagemSEFAZ”, onde:

- » **numeroSessao:** Número aleatório gerado pelo AC para controle da comunicação;



- » **EEEEEE:** O código EEEEE é substituído por um EEEEE dentre os possíveis códigos de retorno descritos no item 6.2 da Especificação (Referência 1);
- » **mensagem:** São as descrições dos códigos EEEEE;
- » **cod:** Código de referência de cada mensagemSEFAZ;
- » **mensagemSEFAZ:** Mensagem de texto enviada pela *Sefaz* referente ao "Envio de avisos ao usuário".

#### 2.1.14 DesbloquearSAT

O Aplicativo Comercial ou outro software fornecido pelo Fabricante poderá realizar o desbloqueio operacional do Equipamento SAT.

Para essa funcionalidade deve-se utilizar a função:

`String DesbloquearSAT (long numeroSessao, String codigoDeAtivacao);`

##### 2.1.14.1 Parâmetros

Os parâmetros são:

- » **numeroSessao[type long]:** Número aleatório gerado pelo AC para controle da comunicação;
- » **codigoDeAtivacao[type String]:** Senha definida pelo contribuinte no software de ativação.

##### 2.1.14.2 Retorno

A função deverá retornar um valor do tipo string contendo os dados separados por pipe "|".

Segue a estrutura do retorno de dados:

**Retorno:** "numeroSessao|EEEEEE|mensagem|cod|mensagemSEFAZ", onde:

- » **numeroSessao:** Número aleatório gerado pelo AC para controle da comunicação;
- » **EEEEEE:** O código EEEEE é substituído por um EEEEE dentre os possíveis códigos de retorno descritos no item 6.2 da Especificação (Referência 1);
- » **mensagem:** São as descrições dos códigos EEEEE;
- » **cod:** Código de referência de cada mensagemSEFAZ;
- » **mensagemSEFAZ:** Mensagem de texto enviada pela *Sefaz* referente ao "Envio de avisos ao usuário".

#### 2.1.15 TrocarCodigoDeAtivacao

Para essa funcionalidade utiliza-se a seguinte função:

```
String TrocarCodigoDeAtivacao (long numeroSessao, String  
codigoDeAtivacao, long opcao, String novoCodigo, String confNovoCodigo);
```

#### **2.1.15.1**      **Parâmetros**

Em caso de troca do código de ativação por esquecimento, o parâmetro “codigoDeAtivacao” deverá ser preenchido com o código de ativação de emergência e o parâmetro "opcao" deve ter o valor "2", caso contrário deve ter o valor "1". Os parâmetros são:

- » **numeroSessao[type long]:** Número aleatório gerado pelo AC para controle da comunicação;
- » **codigoDeAtivacao[type String]:** Senha definida pelo contribuinte no software de ativação;
- » **opcao[type long]:** Refere-se ao conteúdo do parâmetro "codigoDeAtivacao" e segue a relação descrita acima;
- » **novoCodigo[type String]:** Novo código de ativação;
- » **confNovoCodigo[type String]:** Confirmação do novo código de ativação.

#### **2.1.15.2**      **Retorno**

A função deverá retornar um valor do tipo string contendo os dados separados por pipe “|”.

Segue a estrutura do retorno de dados:

**Retorno:** “numeroSessao|EEEEEE|mensagem|cod|mensagemSEFAZ”, onde:

- » **numeroSessao:** Número aleatório gerado pelo AC para controle da comunicação;
- » **EEEEEE:** O código EEEEE é substituído por um EEEEE dentre os possíveis códigos de retorno descritos no item 6.2 da Especificação (Referência 1);
- » **mensagem:** São as descrições dos códigos EEEEE;
- » **cod:** Código de referência de cada mensagemSEFAZ;
- » **mensagemSEFAZ:** Mensagem de texto enviada pela *Sefaz* referente ao "Envio de avisos ao usuário".

### 3 Referências

---

- » Especificação de Requisitos do SAT, Versão ER 2\_20\_06 – 19/08/2016 disponível em [http://www.fazenda.sp.gov.br/sat/downloads/Especificacao SAT v ER 2 20 06.pdf](http://www.fazenda.sp.gov.br/sat/downloads/Especificacao%20SAT%20v%20ER%2020%2006.pdf)

Kryptus Segurança da Informação Ltda.

[sat@kryptus.com](mailto:sat@kryptus.com)

Tel/Fax +55 (19) 3112 5050

<http://www.satcfe.com.br>

