



# Descrição da Solução NAVS

(Navegador Skytef)

## SKYTEF

Versão 1.36.0 – 04/10/2017

## Sumário

<b>1.0</b>	<b>Introdução</b>	<b>6</b>
<b>2.0</b>	<b>Descrição Inicial</b>	<b>7</b>
	<b>Lista de códigos de status HTTP</b>	<b>7</b>
	<b>1xx Informativa</b>	<b>7</b>
	100 Continuar	7
	101 Mudando protocolos	8
	102 Processamento (WebDAV) (RFC 2518)	8
	122 Pedido-URI muito longo	8
	<b>2xx Sucesso</b>	<b>8</b>
	200 OK	8
	201 Criado	8
	202 Aceito	8
	203 não-autorizado (desde HTTP/1.1)	8
	204 Nenhum conteúdo	8
	205 Reset	9
	206 Conteúdo parcial	9
	207-Status Multi (WebDAV) (RFC 4918)	9
	<b>3xx Redirecionamento</b>	<b>9</b>
	300 Múltipla escolha	9
	301 Movido	9
	302 Encontrado	9
	304 Não modificado	9
	305 Use Proxy (desde HTTP/1.1)	10
	306 Proxy Switch	10
	307 Redirecionamento temporário (desde HTTP/1.1)	10
	<b>4xx Erro de cliente</b>	<b>10</b>
	400 Requisição inválida	10
	401 Não autorizado	10
	402 Pagamento necessário	10
	403 Proibido	11
	404 Não encontrado	11
	405 Método não permitido	11
	406 Não Aceitável	11
	407 Autenticação de proxy necessária	11
	408 Tempo de requisição esgotou (Timeout)	11
	409 Conflito	11
	410 Gone	11
	411 comprimento necessário	11
	412 Pré-condição falhou	12
	413 Entidade de solicitação muito grande	12
	414 Pedido-URI Too Long	12
	415 Tipo de mídia não suportado	12
	416 Solicitada de Faixa Não Satisfatória	12
	417 Falha na expectativa	12

## Especificação de Interface

418 Eu sou um bule de chá	12
422 Entidade improcessável (WebDAV) (RFC 4918)	12
423 Fechado (WebDAV) (RFC 4918)	12
424 Falha de Dependência (WebDAV) (RFC 4918)	12
425 coleção não ordenada (RFC 3648)	13
426 Upgrade Obrigatório (RFC 2817)	13
450 bloqueados pelo Controle de Pais do Windows	13
499 cliente fechou Pedido (utilizado em ERPs/VPsA)	13
<b>5xx outros erros</b>	<b>13</b>
500 Erro interno do servidor (Internal Server Error)	13
501 Não implementado (Not implemented)	13
502 Bad Gateway	14
503 Serviço indisponível (Service Unavailable)	14
504 Gateway Time-Out	14
505 HTTP Version not supported	14
<b>3.0 Detalhamento dos Comandos</b>	<b>15</b>
<b>3.1 CONSOLE</b>	<b>16</b>
<b>3.2 GET</b>	<b>18</b>
3.2.1 CARTÃO MAGNÉTICO	19
3.2.2 ENTRADA NUMÉRICA	20
3.2.3 ENTRADA ALFANUMÉRICA	21
3.2.4 ENTRADA NUMÉRICA SECRETA	22
3.2.5 ENTRADA NUMÉRICA COM CASAS DECIMAIS	23
3.2.6 MÁSCARA GENÉRICA	24
3.2.7 AGUARDA TECLA ESPECIAL	26
3.2.8 AGUARDA TECLA	27
3.2.9 STATUS ATUAL DA COMUNICAÇÃO NO POS	28
3.2.1 STATUS ATUAL DA BATERIA EM PERCENTUAL	29
3.2.2 CONTROLA SESSÃO	29
3.2.3 VALOR MOEDA	30
3.2.4 NUMERO DE SÉRIE	31
3.2.5 CÓDIGO DE BARRAS	32
3.2.6 PORTA SERIAL	33
3.2.7 CONTROLES DA PORTA SERIAL	34
3.2.8 VERSÃO	35
3.2.9 VERSÃO DO SISTEMA OPERACIONAL	36
3.2.10 MODELO	37
3.2.11 IDENTIFICAÇÃO DO CHIP	38
3.2.12 MIFARE	39
3.2.13 CTLS	39
3.2.14 LEITURA DE DADOS DO MIFARE	40
3.2.15 TOUCH (somente no Vx680)	41
3.2.16 DATA E HORA	43
3.2.17 CPF	44
3.2.18 HOUR	45
3.2.19 DATA	46
3.2.20 CNPJ	47
3.2.21 APDUCHIP	48
3.2.22 DESNBS	49
3.2.23 INFO_FILE	50
3.2.24 DRIVE_SIZE	51
3.2.25 EMV	52

## Especificação de Interface

3.3	DELAY	53
3.4	SELECT	54
3.5	FTP	56
3.6	FINGERPRINT	58
3.7	TONE	60
3.8	PRINTER	61
3.9	CANCEL_KEY	61
3.10	CHGCONFNT	62
3.11	PRNFNT	63
3.12	WRITE_AT	64
3.13	DRAW (Somente para Vx680)	65
3.14	CHGPRNFNT	66
3.15	PRNLOGO	68
3.16	CONLOGO	69
3.17	ANIMATE	70
3.18	CODEBAR – Impressão	71
3.19	GERADOR DE QR CODES	72
3.20	Criar Código de barras 3 de 9	73
3.21	Criar Código de Barras ITF (2 de 5)	74
3.22	Criar Código de Barras CODE-128	75
3.23	SAÍDA SERIAL	76
3.24	SET TIME	76
3.25	SET BACKLIGHT	77
3.26	SET KBBACKLIGHT	78
3.27	KEYBOARD	79
3.28	SET PORTA SERIAL	81
3.29	CAPTURA ÚNICA DE ENTRADA	82
3.30	INIT	84
3.31	POST	84
3.32	FILE	86
3.33	REMOVE ARQUIVO NO POS	89
3.34	DUAL CHIP	90
3.35	COPIA DE/PARA DISPOSITIVOS EXTERNOS - ARQUIVOS	91

## Especificaão de Interface

3.36	RESET	92
3.37	CONFIG_NAVS	93
3.38	IMPRESSÃO DE BOLETO	100
3.39	ENVIO DE ARQUIVO	102
3.40	UNZIP DE ARQUIVO NO POS	103
3.41	COMANDOS AND, OR, XOR, URL_encode e URL_decode para LUA	104
3.42	HTMLDEBUG	105
3.43	RELICENCIA	106
3.44	GRAVAÇÃO EM ARQUIVO DAS CAPTURAS – POSTOFF (uso para offline)	107
3.45	IMPRESSÃO OFFLINE – PRINTEROFF	108
3.46	VALIDAÇÃO OFFLINE – VALIDATEOFF	109
3.47	TELECARGA	111
4.0	OFFLINE com Lua / SQLite	112
5.0	Compilador Lua	115
6.0	MIFARE – leitura e escrita	116
7.0	PAGAMENTO	117
8.0	WEB SERVICES	128
9.0	CRIPTOGRAFIA	129
10.0	GPS	130
11.0	Descrião da Mensagem do POS ao Servidor	132
12.0	Exemplo de uso	133
13.0	Configuraão do NAVS	135
14.0	Economia de energia	147
15.0	Status do Terminal	148
16.0	Histórico das Versões	150

### 1.0 Introdução

NAVS é a plataforma de desenvolvimento da Skytef que proporciona às redes de POS flexibilidade nas aplicações. Possibilita rápido desenvolvimento de novas funcionalidades e serviços, atendendo aos mais diversos mercados como redes adquirentes, *private label*, bancos, financeiras, correspondente bancário, fidelidade, entre outros.

NAVS é um navegador que permite que o POS acesse o servidor, utilizando o mesmo tipo de conexão das transações. O acesso ao servidor possibilita realizar atualizações das aplicações e implementar novos serviços, o que pode ser feito em toda rede ou em um único terminal, de acordo com a necessidade do gestor.

#### Vantagens

- Reduz tempo nos processos de certificação de aplicações
- Transforma a rede de POS numa rede de serviços
- Possibilita ativação/desativação de serviços nos terminais POS remotamente
- Facilita o desenvolvimento, pois é uma plataforma amigável e intuitiva

## 2.0 Descrição Inicial

NAVS é uma aplicação que será carregada no POS e interpretará comandos recebidos do Servidor.

Na carga inicial, deverá existir um arquivo XML contendo as informações da primeira comunicação do POS para o Servidor, como uma inicialização.

Na resposta, o Servidor deverá enviar os comandos XML para terminar a inicialização e iniciar a primeira transação.

O POS capturará os dados desta transação e enviará ao Servidor que deverá responder com os comandos para terminar a transação e iniciar a próxima.

### IMPORTANTE:

O NAVS utiliza o protocolo HTTP ou HTTPS.

O campo mandatório que deve ser enviado pelo Host no protocolo HTTP é o "Content\_Lenght".

Abaixo os principais erros do protocolo HTTP, conforme a Wikipedia:

([https://pt.wikipedia.org/wiki/Lista\\_de\\_c%C3%B3digos\\_de\\_estado\\_HTTP](https://pt.wikipedia.org/wiki/Lista_de_c%C3%B3digos_de_estado_HTTP)).

### *Lista de códigos de status HTTP*

A seguir está uma lista de códigos de resposta em HTTP (HyperText Transfer Protocol). Isso inclui os códigos padrões de internet da IETF, outras especificações e alguns códigos adicionais usados. O primeiro dígito do código de status indica uma das cinco classes de resposta, o mínimo necessário para um cliente HTTP é que ele reconheça essas cinco classes. Microsoft IIS pode usar sub-códigos decimais adicionais específicos para fornecer mais informações, mas estes não estão listados aqui. As frases utilizadas são os exemplos padrão, mas qualquer alternativa humana legível pode ser fornecida. Salvo disposição em contrário, o código de status é parte do padrão HTTP/1.1.

### *1xx Informativa*

Solicitação recebida, dando continuidade ao processo. Essa classe de códigos de status indica uma resposta provisória, constituída apenas do Status-Line e cabeçalhos opcionais, e é encerrado por uma linha vazia. Desde a versão HTTP/1.0 não são definidos todos os códigos de status 1xx, os servidores não devem enviar uma resposta 1xx para um cliente HTTP/1.0, exceto sob condições experimentais.

## 100 Continuar

Isso significa que o servidor recebeu os cabeçalhos da solicitação, e que o cliente deve proceder para enviar o corpo do pedido (no caso de haver um pedido um corpo deve ser enviado, por exemplo, um POST pedido). Se o corpo é grande o pedido, enviando-os para um servidor, quando o pedido já foi rejeitada com base em cabeçalhos inadequado é ineficiente. Para ter um cheque do servidor se o pedido pode ser aceite com base no pedido de cabeçalhos sozinho, o cliente deve enviar Esperar: 100-continue como um cabeçalho no seu pedido

## **Especificação de Interface**

---

inicial e verifique se a 100 Continuar código de status é recebido em resposta antes de permanente (ou receber 417 Falha na expectativa e não continuar).

### **101 Mudando protocolos**

Isso significa que o solicitante pediu ao servidor para mudar os protocolos e o servidor está reconhecendo que irá fazê-lo.

### **102 Processamento (WebDAV) (RFC 2518)**

Como uma solicitação WebDAV pode conter muitos sub-pedidos que envolvam operações de arquivo, pode demorar muito tempo para concluir o pedido. Este código indica que o servidor recebeu e está processando o pedido, mas nenhuma resposta ainda não está disponível. Isso impede que o cliente o tempo limite e supondo que o pedido foi perdido.

### **122 Pedido-URI muito longo**

Este é um padrão IE7 somente código não significa que o URI é mais do que um máximo de 2083 caracteres. (Ver código 414).

#### *2xx Sucesso*

---

Esta classe de códigos de status indica a ação solicitada pelo cliente foi recebida, compreendida, aceita e processada com êxito.

### **200 OK**

Padrão de resposta para solicitações HTTP sucesso. A resposta real dependerá do método de solicitação usado. Em uma solicitação GET, a resposta conterá uma entidade que corresponde ao recurso solicitado. Em uma solicitação POST a resposta conterá a descrição de uma entidade, ou contendo o resultado da ação.

### **201 Criado**

O pedido foi cumprido e resultou em um novo recurso que está sendo criado.

### **202 Aceito**

O pedido foi aceito para processamento, mas o tratamento não foi concluído. O pedido poderá ou não vir a ser posta em prática, pois pode ser anulado quando o processamento ocorre realmente.

### **203 não-autorizado (desde HTTP/1.1)**

O servidor processou a solicitação com sucesso, mas está retornando informações que podem ser de outra fonte.

### **204 Nenhum conteúdo**

O servidor processou a solicitação com sucesso, mas não é necessário nenhuma resposta.



## 205 Reset

O servidor processou a solicitação com sucesso, mas não está retornando nenhum conteúdo. Ao contrário da 204, esta resposta exige que o solicitante redefinir a exibição de documento.

## 206 Conteúdo parcial

O servidor está entregando apenas parte do recurso devido a um cabeçalho intervalo enviados pelo cliente. O cabeçalho do intervalo é usado por ferramentas como wget para permitir retomada de downloads interrompidos, ou dividir um download em vários fluxos simultâneos.

## 207-Status Multi (WebDAV) ([RFC 4918](#))

O corpo da mensagem que se segue é um XML da mensagem e pode conter um número de códigos de resposta individual, dependendo de quantas sub-pedidos foram feitos.

### *3xx Redirecionamento*

---

O cliente deve tomar medidas adicionais para completar o pedido. Essa classe de código de status indica que a ação ainda precisa ser levado pelo agente do usuário, a fim de atender à solicitação. A ação necessária pode ser realizada pelo agente, sem interação com o usuário, se e somente se o método utilizado no segundo pedido é GET ou HEAD. Um agente do usuário não deve redirecionar automaticamente uma solicitação de mais de cinco vezes, uma vez que tais redirecionamentos geralmente indicam um loop infinito .

## 300 Múltipla escolha

Indica várias opções para o recurso que o cliente pode acompanhar. É, por exemplo, poderia ser usado para apresentar opções de formato diferente para o vídeo, arquivos de lista com diferentes extensões, ou desambiguação sentido da palavra.

## 301 Movido

Esta e todas as solicitações futuras devem ser direcionada para o URI.

## 302 Encontrado

Este é um exemplo de boas práticas industriais contradizendo a norma. Especificação HTTP/1.0 ([RFC 1945](#)) exigiu o cliente para executar um redirecionamento temporário (o que descreve frase original era “Movido Temporariamente”), mas os browsers populares executadas 302 com a funcionalidade de um 303 Consulte Outros. Por isso, acrescentou HTTP/1.1 códigos de status 303 e 307 a distinguir entre os dois comportamentos. No entanto, a maioria das aplicações Web e os quadros ainda usam o código de status 302 como se fosse o 303.

## 304 Não modificado

Indica que o recurso não foi modificado desde o último pedido. Normalmente, o cliente fornece um cabeçalho HTTP como o Se-Modificado-Desde cabeçalho para proporcionar um tempo contra o qual para comparar.

## **Especificação de Interface**

---

Usando este poupa largura de banda e de reprocessamento no servidor e cliente, uma vez que apenas os dados do cabeçalho devem ser enviados e recebidos em comparação com a totalidade da página que está sendo reprocessados pelo servidor, em seguida, enviado novamente utilizando mais largura de banda do servidor e cliente.

### **305 Use Proxy (desde HTTP/1.1)**

Muitos clientes HTTP (como o Mozilla e Internet Explorer) podem não tratar corretamente as respostas com este código de status, principalmente por razões de segurança.

### **306 Proxy Switch**

Mudança de proxy. Deixou de ser usado.

### **307 Redirecionamento temporário (desde HTTP/1.1)**

Nesta ocasião, o pedido deve ser repetido com outro URI, mas futuras solicitações ainda pode usar a URI original. Em contraste com a 303, o método de pedido não deve ser mudado quando a reedição do pedido original. Por exemplo, uma solicitação POST deve ser repetido com outro pedido POST.

## ***4xx Erro de cliente***

---

A classe 4xx de código de status é destinado para os casos em que o cliente parece ter cometido um erro. Exceto quando estiver respondendo a uma solicitação HEAD, o servidor deve incluir uma entidade que contém uma explicação sobre a situação de erro, e se é uma condição temporária ou permanente. Esses códigos de status são aplicáveis a qualquer método de solicitação. Os agentes do usuário devem exibir qualquer entidade incluída para o usuário. Estes são tipicamente os códigos de erro mais comuns encontrados durante online.

### **400 Requisição inválida**

O pedido não pode ser entregue devido à sintaxe incorreta.

### **401 Não autorizado**

Semelhante ao 403 Proibido, mais especificamente para o uso quando a autenticação é possível, mas não conseguiu ou ainda não foram fornecidos. A resposta deve incluir um cabeçalho do campo www-authenticate contendo um desafio aplicável ao recurso solicitado. Veja Basic autenticação de acesso e autenticação Digest acesso.

### **402 Pagamento necessário**

Reservado para uso futuro. A intenção original era que esse código pudesse ser usado como parte de alguma forma de dinheiro digital ou de micro pagamento regime, mas isso não aconteceu, e esse código não é usado normalmente.

### **403 Proibido**

O pedido é reconhecido pelo servidor mas este recusa-se a executá-lo. Ao contrário resposta “401 Não Autorizado”, autenticação não fará diferença e o pedido não deve ser requisitado novamente.

### **404 Não encontrado**

O recurso requisitado não foi encontrado, mas pode ser disponibilizado novamente no futuro. As solicitações subsequentes pelo cliente são permitidas.

### **405 Método não permitido**

Foi feita uma solicitação de um recurso usando um método de pedido não é compatível com esse recurso, por exemplo, usando GET em um formulário, que exige que os dados a serem apresentados via POST, PUT ou usar em um recurso somente de leitura.

### **406 Não Aceitável**

O recurso solicitado é apenas capaz de gerar conteúdo não aceitáveis de acordo com os cabeçalhos Accept enviados na solicitação.

### **407 Autenticação de proxy necessária**

### **408 Tempo de requisição esgotou (Timeout)**

O servidor sofreu timeout ao aguardar a solicitação. De acordo com as especificações HTTP W3: “O cliente não apresentou um pedido dentro do tempo que o servidor estava preparado para esperar. O cliente PODE repetir o pedido sem modificações a qualquer momento mais tarde.”

### **409 Conflito**

Indica que a solicitação não pôde ser processada por causa do conflito no pedido, como um conflito de edição.

### **410 Gone**

Indica que o recurso solicitado não está mais disponível e não estará disponível novamente. Isto deve ser usado quando um recurso foi intencionalmente removido e os recursos devem ser removidos. Ao receber um código de estado 410, o cliente não deverá solicitar o recurso novamente no futuro. Clientes como motores de busca devem remover o recurso de seus índices. A maioria dos casos de uso não necessitam de clientes e motores de busca para purgar o recurso, e um “404 Not Found” pode ser utilizado.

### **411 comprimento necessário**

O pedido não especifica o comprimento do seu conteúdo, o que é exigido pelo recurso solicitado.

### 412 Pré-condição falhou

O servidor não cumpre uma das condições que o solicitante coloca na solicitação.

### 413 Entidade de solicitação muito grande

A solicitação é maior do que o servidor está disposto ou capaz de processar.

### 414 Pedido-URI Too Long

O URI fornecido foi muito longo para ser processado pelo servidor.

### 415 Tipo de mídia não suportado

A entidade tem um pedido tipo de mídia que o servidor ou o recurso não tem suporte. Por exemplo, o cliente carrega uma imagem como image / svg + xml, mas o servidor requer que imagens usar um formato diferente.

### 416 Solicitada de Faixa Não Satisfatória

O cliente solicitou uma parte do arquivo, mas o servidor não pode fornecer essa parte. Por exemplo, se o cliente pediu uma parte do arquivo que está para além do final do arquivo.

### 417 Falha na expectativa

O servidor não pode cumprir as exigências do campo de cabeçalho Espere-pedido.

### 418 Eu sou um bule de chá

Este código foi definido em 1998 como uma das tradicionais brincadeiras de 1º de abril da IETF, na [RFC 2324](#), Hyper Text Cafeteira Control Protocol, e não é esperado para ser implementado por servidores HTTP reais.

### 422 Entidade improcessável (WebDAV) ([RFC 4918](#))

O pedido foi bem formado, mas era incapaz de ser seguido devido a erros de semântica.

### 423 Fechado (WebDAV) ([RFC 4918](#))

O recurso que está sendo acessado está bloqueado.

### 424 Falha de Dependência (WebDAV) ([RFC 4918](#))

A solicitação falhou devido à falha de uma solicitação anterior (por exemplo, um PROPPATCH).

### 425 coleção não ordenada ([RFC 3648](#))

Definido em projectos de “WebDAV Avançada Coleções Protocolo”, mas não está presente no “Web Distributed Authoring and Versioning (WebDAV) Ordenados Coleções protocolo”.

### 426 Upgrade Obrigatório ([RFC 2817](#))

O cliente deve mudar para um outro protocolo, como TLS/1.0 . Resposta n.º 444 Um Nginx extensão do servidor HTTP. O servidor retorna nenhuma informação para o cliente e fecha a conexão (útil como um impedimento para malware). Com 449 Repetir Uma extensão de Microsoft. O pedido deve ser repetida após a realização da ação apropriada.

### 450 bloqueados pelo Controle de Pais do Windows

Uma extensão de Microsoft. Este erro é dado quando Parental Controls do Windows estão ativadas e está bloqueando o acesso a determinada página da web.

### 499 cliente fechou Pedido (utilizado em ERPs/VPs)

Um Nginx extensão do servidor HTTP. Este código é introduzido para registrar o caso quando a conexão é fechada pelo cliente ao servidor HTTP é o processamento de seu pedido, fazendo com que servidor não consiga enviar o cabeçalho HTTP de volta.

## *5xx outros erros*

---

### 500 Erro interno do servidor (Internal Server Error)

Indica um erro do servidor ao processar a solicitação. Na grande maioria dos casos está relacionada as permissões dos arquivos ou pastas do software ou script que o usuário tenta acessar e não foram configuradas no momento da programação/construção do site ou da aplicação. Para corrigir, verifique o diretório em que o arquivo ou recurso que houve falha de acesso está localizado, e este arquivo (bem como todos os outros), obedeçam às regras seguintes:

Pastas — chmod 755 (não utilizar 777) Arquivos — chmod 644 (não utilizar o 777, só utilizar outro se for expressamente solicitado na instalação)

OBS.: algumas aplicações e ou sistemas requerem permissões diferenciadas, pelo qual é importante verificar com os criadores do scripts/sistema, qual seria a permissão correta a usar. O exemplo descreve como é realizado em sistemas operacionais Unix-like. Fazer analogia como é realizado em sistemas como Windows (Windows 7, 8, XP entre outros).

Este erro também pode ocorrer se o arquivo .htaccess do seu site estiver modificando os parâmetros ou tentando fazer o PHP utilizar comandos como: php\_flag ou php\_value. Remova qualquer entrada com esses comandos do arquivo .htaccess. Se for fazer modificações nos parâmetros do PHP, utilize o arquivo php.ini para fazer isso.

### 501 Não implementado (Not implemented)

O servidor ainda não suporta a funcionalidade ativada

### 502 Bad Gateway

Em regra, o erro quando há uma configuração imprecisa entre os computadores de back-end, possivelmente incluindo o servidor Web no site visitado. Antes de analisar este problema, é necessário limpar o cache do navegador, completamente.

Se estiver navegando na Web e observar este problema em todos os websites visitados, então 1) o seu provedor de serviço de Internet tem uma falha/sobrecarga em um equipamento principal ou 2) tem algo de errado com a sua conexão interna à Internet, por exemplo, o firewall não está funcionando corretamente. Se for o primeiro caso, somente o seu provedor pode ajudar. Se for o segundo, você precisa corrigir o que quer que esteja prevenindo que você acesse a Internet.

Se tiver este problema somente em alguns websites visitados, provavelmente existe um problema nos sites. Por exemplo, uma das peças dos equipamentos estão falhando ou estão sobrecarregadas. Entre em contato com os responsáveis destes sites.

### 503 Serviço indisponível (Service Unavailable)

O servidor está em manutenção ou não consegue dar conta dos processamentos de recursos devido à sobrecarga do sistema. Isto deve ser uma condição temporária.

### 504 Gateway Time-Out

É caracterizado por erros particulares do site em questão. Pode ser que o site esteja em manutenção ou não exista.

### 505 HTTP Version not supported

A maioria dos browsers assumem que os servidores de rede suportam versões 1.x do protocolo HTTP. Na prática, as versões muito antigas como a 0.9 são pouco utilizadas atualmente, não apenas porque eles fornecem pouca segurança e desempenho mais baixo do que as versões mais recentes do protocolo. Então, se acontecer esse erro no seu navegador de rede, a única opção é fazer o upgrade do software do servidor de rede. Se a versão da solicitação 1.x falhar, pode ser porque o servidor de rede está suportando versões incorretas do protocolo 1.x, em vez de não suportá-las.

6xx outros erros

### 3.0 Detalhamento dos Comandos

Através de 'scripts', o servidor designará os comandos a serem executados pelo POS, como display, impressora, cartão, captura de dados, etc.

Obs.: Quando tiver variáveis nos comandos para receber algum retorno, não coloque variáveis no seu código com o mesmo nome.

Exemplo: se o comando for `<GET NAME=CAMPO1 TYPE=VALUE SIZE=X COL=Y LIN=Z>`, não coloque uma variável NAME para NAME (~~`<GET NAME=NAME TYPE=VALUE SIZE=X COL=Y LIN=Z>`~~).

Obs2.: Para os comandos que utilizam os parâmetros LINHA e COLUNA na tela do POS, é importante ressaltar que linha e coluna iniciais possuem inicialmente os respectivos valores : 1 e 0, portanto se quiser posicionar algo na primeira linha e na primeira coluna no console devemos atribuir 1 para linha e 0 para coluna.

A seguir o detalhamento de alguns 'scripts':

## Especificação de Interface

### 3.1 CONSOLE

<CONSOLE NOCLS=x> inicia texto para o display,  
</CONSOLE> finaliza texto para o display.

A mudança de linha é realizada pelo comando <BR>.

O parâmetro NOCLS determina se uma imagem que estiver no display será apagada ou não com a atualização do texto na tela.

X representa os valores que podem ser assumidos 0=não 1=sim.

No Vx520, caso a mensagem ultrapassar o número de linhas da Console, use as teclas de Scroll (definidas no menu de configuração), para navegar pelas linhas.

Obs.: Qualquer outro comando deverá ser utilizado fora de "<CONSOLE NOCLS=x>yyy</CONSOLE>", entre eles haverá SOMENTE texto para o display.

Obs.2: Caracteres especiais devem ser passados com o respectivo número hexadecimal para que possam ser interpretados corretamente e ter a exibição desejada na tela. Para exibir na tela o caractere de seta para cima utilize "\x14\x15", para exibir seta para baixo utilize "\x16\x17", para exibir seta para direita utilize "\x18", e exibir seta para esquerda utilize "\x19".

Abaixo um exemplo de como inserir seta para cima no console :

<CONSOLE NOCLS=0>\x14\x15</CONSOLE>

A seguir tabelas com alguns valores correspondentes em hexadecimal de caracteres especiais:

Caractere	Valor Hexadecimal correspondente
!	21
"	22
#	23
\$	24
%	25
&	26
'	27
(	28
)	29
*	2a
+	2B
,	2C
-	2D
÷	F7
.	2E
:	3a
;	3B
=	3D
>	3E

Caractere	Valor Hexadecimal correspondente
?	3F
@	40
[	5B
\	5C
]	5D
^	5E
`	60
{	7B
	7C
}	7D
~	7E
ñ	F1
✓	8B
ı	A1
¢	A2
£	A3
¥	A5
§	A7
©	A9



## Especificação de Interface

Caractere	Valor Hexadecimal correspondente
¬	AC
ë	EB
®	AE
°	B0
à	E0
á	E1
â	E2
ã	E3
ä	E4
å	E5
μ	B5
¶	B6
¿	BF
À	C0
Á	C1
Â	C2
Ã	C3

Caractere	Valor Hexadecimal correspondente
Æ	C6
Ç	C7
È	C8
É	C9
Ê	CA
Ì	CC
Í	CD
Î	CE
Ï	CF
Ñ	D1
æ	E6
ç	E7
è	E8
é	E9
ê	EA
Ä	C4
Å	C5

\*É importante salientar que esses valores são válidos apenas para a fonte "DEFAULT" dos POS Verifone (Vx520 e Vx680). Para outras fontes Verifone e aparelhos Ingenico a correspondência relacionada nas tabelas acima não é garantida.

### 3.2 GET

<GET...> indica a captura de informações no POS, a seguir os parâmetros mandatórios deste comando:

TYPE discrimina o tipo da informação a ser capturada.

NAME discrimina o identificador do campo que contém a informação capturada.

Obs.: A partir da versão 1.17.0, ocupa apenas os espaços do texto escrito (não apaga mais a linha toda).  
Podendo por exemplo usar da seguinte forma na mesma linha:

```
<CONSOLE>  
DIGITE O TIPO:  
</CONSOLE>  
<GET TYPE=FIELD NAME=TIPO LIN=1 COL=16 SIZE=3>
```

Na tela do POS após digitar 123, ficará:

DIGITE O TIPO: 123

### 3.2.1 CARTÃO MAGNÉTICO

Para capturar os dados do cartão, através da leitura da tarja magnética, devemos passar para o parâmetro TYPE o argumento CARD.

Neste caso, haverá outro parâmetro mandatório: TRACKS, que especifica quais trilhas do cartão deverão ser coletadas.

O comando deve ser passado conforme o esquema abaixo, notando que a ordem dos parâmetros não altera o resultado.

<GET TYPE=CARD NAME=CARTAO1 TRACKS=X>

'X' pode ser qualquer valor de 0 a 7, da seguinte forma:

0-não envia nenhuma trilha.

1-envia trilha 1

2-envia trilha 2

4-envia trilha 3

3-envia trilhas 1 e 2

5-envia trilhas 1 e 3

6-envia trilhas 2 e 3

7-envia trilhas 1, 2 e 3

**IMPORTANTE:** As informações do cartão são enviadas ao Servidor usando "URLENCODE":

Retorna uma string em que todos os caracteres não-alfanuméricos com exceção de **-\_.** São codificados como um sinal de por cento (**%**) seguido por dois dígitos hexadecimais e espaços codificados como um sinal de (**+**).

### 3.2.2 ENTRADA NUMÉRICA

Para capturar uma entrada numérica, não secreta, por meio do teclado devemos passar para o parâmetro TYPE o argumento FIELD.

Neste caso haverá os parâmetros mandatórios: SIZE, COL, LIN.  
O parâmetro NOENTER é opcional.

SIZE especifica o tamanho em caracteres do campo a ser capturado.

COL especifica a coluna da esquerda para direita onde o primeiro caractere capturado será mostrado no display.

LIN especifica a linha de cima para baixo onde o primeiro caractere capturado será mostrado no display.

NOENTER=1 especifica que o comando não aguardará o <ENTER> no final da entrada.

ZL se esse parâmetro estiver listado, significa que os zeros que forem digitados antes de outro número serão ignorados.

Devemos notar que na linha onde será mostrado o campo, todo texto previamente existente será apagado.

O comando deverá ser passado conforme o esquema abaixo:

```
<GET TYPE=FIELD NAME=CAMPO1 SIZE=X COL=Y LIN=Z NOENTER=1 ZL>
```

### 3.2.3 ENTRADA ALFANUMÉRICA

Para capturar uma entrada alfanumérica, não secreta, por meio do teclado devemos passar os mesmos argumentos da Entrada Numérica acima, incluindo o parâmetro ALPHA=1, como abaixo:

<GET TYPE=FIELD ALPHA=1 NAME=CAMPO1 SIZE=X COL=Y LIN=Z NOENTER=1 ZL>

### 3.2.4 ENTRADA NUMÉRICA SECRETA

Para capturar uma entrada numérica, secreta, por meio do teclado devemos passar para o parâmetro TYPE o argumento PASS.

Neste caso haverá os parâmetros mandatórios: SIZE, COL, LIN.

O parâmetro NOENTER é opcional.

SIZE especifica o tamanho em caracteres do campo a ser capturado.

COL especifica a coluna da esquerda para direita onde o primeiro asterisco referente ao primeiro caractere capturado será mostrado no display.

LIN especifica a linha de cima para baixo onde o primeiro asterisco referente ao primeiro caractere capturado será mostrado no display.

ZL se esse parâmetro estiver listado, significa que os zeros que forem digitados antes de outro número serão ignorados.

NOENTER=1 especifica que o comando não aguardará o <ENTER> no final da entrada.

Devemos notar que na linha onde será mostrado o campo, todo texto previamente existente será apagado.

O comando deverá ser passado conforme o esquema abaixo:

```
<GET TYPE=PASS NAME=CAMPO2 SIZE=X COL=Y LIN=Z NOENTER=1 ZL>
```

### 3.2.5 ENTRADA NUMÉRICA COM CASAS DECIMAIS

Para capturar uma entrada numérica, não secreta, e que tenha a formatação de casas decimais, por meio do teclado devemos passar para o parâmetro TYPE o argumento NONINT.

Neste caso haverá os parâmetros mandatórios: SIZE, COL, LIN, DECIMALS.  
O parâmetro NOENTER é opcional.

SIZE especifica o tamanho em caracteres do campo a ser capturado.

COL especifica a coluna da esquerda para direita onde o primeiro caractere capturado será mostrado no display.

LIN especifica a linha de cima para baixo onde o primeiro caractere capturado será mostrado no display.

DECIMALS especifica o número de casas decimais.

NOENTER especifica que o comando não aguardará o <ENTER> no final da entrada.

ZL se esse parâmetro estiver listado, significa que os zeros que forem digitados antes de outro número serão ignorados.

Devemos notar que na linha onde será mostrado o campo, todo texto previamente existente será apagado.

O comando deverá ser passado conforme o esquema abaixo:

```
<GET TYPE=NONINT NAME=CAMPO1 SIZE=X COL=Y LIN=Z DECIMALS=W NOENTER=1 ZL>
```

### 3.2.6 MÁSCARA GENÉRICA

Para captura entrada com máscara pré-definida na tela.

O valor retornado virá sem a máscara.

Esse comando acaba substituindo muitos outros GET, mas eles serão mantidos por questão de compatibilidade.

Sintaxe:

<GET TYPE=FIELD NAME=nome LIN=lin COL=col MASK=mascara LABEL=rotulo NOENTER=1 ZL>

Onde:

nome = nome da variável que receberá o valor digitado

lin = linha onde será posicionado o campo.

Col = coluna onde será posicionado o campo.

Rotulo = texto a ser colocado à frente do campo a entrar, quando o AutoClear estiver ativo.

Mascara = mascara de digitação conforme a regra abaixo:

NOENTER=1 especifica que o comando não aguardará o <ENTER> no final da entrada (opcional).

Na máscara podem ser usados os caracteres:

- 9, 0, x, X, a e A, com as seguintes funções:
- 9 – Só aceita dígitos numéricos.
- 0 – Só aceita dígitos numéricos e coloca zero onde ainda não estiver preenchido.
- a – Só aceita caracteres alfabéticos, maiúsculos ou minúsculos.
- A – Só aceita caracteres alfabéticos, forçando para maiúsculos.
- x – Aceita qualquer caractere.
- X – Aceita qualquer caractere, forçando os alfabéticos para maiúsculos.

Alguns caracteres especiais podem ser usados na primeira posição da máscara, para alterar algumas características da entrada de dados:

- \* - Digitação protegida. Todos os caracteres digitados aparecerão como asteriscos.
- ! – Enter automático. Ao se digitar o último caractere da máscara, é gerado um Enter.
- D – A digitação é alinhada à direita.
- E – A digitação é alinhada à esquerda (default).
- \_ - O espaço para entrada do valor é preenchido com o caractere underline.
- C – AutoClear – Ao editar um campo com um valor já definido, caso a primeira tecla digitada seja uma tecla numérica, o campo é limpo. Caso contrario (um BACKSPACE, ou ENTER, por exemplo) o valor do campo é preservado.
- @ - Flip-flop. Neste caso, o usuário deverá escolher um em uma lista de valores. Os valores possíveis deverão vir imediatamente após o '@', separados por vírgula.



## Especificação de Interface

---

Exemplos:

Digitação de senha: “\*999999”

Digitação de data: “99/99/9999”

Escolha de opções: “@Um,Dois,Três,Quatro”

Qualquer outro caractere que apareça na máscara será apresentado na posição onde aparece.

### 3.2.7 AGUARDA TECLA ESPECIAL

Para aguardar o pressionamento de teclas especiais, devemos passar para o parâmetro TYPE o argumento SPECIALKEY, conforme demonstrado a seguir :

<GET TYPE=SPECIALKEY NAME=CAMPO1 NOENTER=1 NOBKSPC=1 >

Os valores retornados pelas teclas são os relacionados na tabela a seguir :

TECLA	STRING RETORNADA
F1	FA
F2	FB
F3	FC
F4	FD
Roxa 1	E1
Roxa 2	E2
ALPHA	8F
Roxa 3	E3
Roxa 4	E4
*	AA
#	A3
BackSpace (Amarela)	88
Confirma (Verde)	8D

Obs.: Nos modelos VX680 da Verifone somente as teclas “BackSpace” e “Confirma” podem ser utilizadas.

Os parâmetros NOENTER e NOBKSPC com valor 1, desabilitam as teclas ENTER e BACK SPACE, respectivamente.

### **3.2.8 AGUARDA TECLA**

Para aguardar o pressionamento da tecla Verde, sem enviar qualquer informação ao Servidor, devemos passar para o parâmetro TYPE o argumento ANYKEY.

Note que neste caso o parâmetro NAME não deverá constar no comando.

<GET TYPE=ANYKEY>

### 3.2.9 STATUS ATUAL DA COMUNICAÇÃO NO POS

Para saber o status da comunicação basta usar esse comando que ele retornará uma string.

O conteúdo retornado é o mesmo que aparece no lado direito da linha de status.

A seguir a forma de utilização do comando :

<GET TYPE=STSCOMM NAME=xxx>, onde :

NAME deverá receber o nome da variável que receberá o status atual de comunicação do POS.

As strings retornadas são as seguintes destacadas em negrito e explicadas logo em seguida:

<b>CX..:</b>	Conectando na rede.
<b>Rdok:</b>	Rede OK, indica conexão física com sucesso.
<b>StNw:</b>	Inicia conexão lógica.
<b>Cxok:</b>	Conexão lógica com sucesso, conectado com a operadora.
<b>Wfok:</b>	Encontrou a rede Wi-Fi.
<b>Htok:</b>	Conectado com o Servidor.
<b>HOST:</b>	Servidor OK.
<b>Hoff:</b>	Servidor offline.
<b>RX..:</b>	Recebendo dados.
<b>Rxok:</b>	Dados do Servidor recebidos.
<b>TX..:</b>	Enviando dados ao Servidor.
<b>CXer:</b>	Erro de conexão.
<b>DNSe:</b>	Erro de DNS, tentando encontrar página.
<b>....:</b>	Reconectando na rede.

### 3.2.1 STATUS ATUAL DA BATERIA EM PERCENTUAL

Para saber o status da bateria basta usar esse comando que ele retornará uma string, com o percentual da bateria.

Os valores são de 0 a 100.

A seguir a forma de utilização do comando :

<GET TYPE=BATTERY\_STATUS NAME=xxx>, onde :

NAME deverá receber o nome da variável que receberá o status atual da bateria do POS.

### 3.2.2 CONTROLA SESSÃO

Para controlar a sessão da comunicação, devemos passar para o parâmetro TYPE o argumento HIDDEN. Os parâmetros NAME e VALUE são sempre obrigatórios.

Assim, os parâmetros NAME=xxx VALUE=yyy são sempre retornados pelo POS.

<GET TYPE=HIDDEN NAME=xxx VALUE=yyy>

Obs.: o tamanho máximo para o parâmetro VALUE é 30 dígitos.

### 3.2.3 VALOR MOEDA

Para capturar um valor moeda numérica, não secreta, por meio do teclado devemos passar para o parâmetro TYPE o argumento VALUE.

Neste caso haverá os parâmetros mandatórios: SIZE, COL, LIN.

ZL se esse parâmetro estiver listado, significa que os zeros que forem digitados antes de outro número serão ignorados.

Será mostrado com R\$, 2 casas decimais e separação com pontos.

Ex.: R\$1.000.000,00.

SIZE especifica o tamanho em caracteres do campo a ser capturado.

COL especifica a coluna da esquerda para direita onde o primeiro caractere capturado será mostrado no display.

LIN especifica a linha de cima para baixo onde o primeiro caractere capturado será mostrado no display.

NOENTER=1 especifica que o comando não aguardará o <ENTER> no final da entrada (opcional).

Devemos notar que na linha onde será mostrado o campo, todo texto previamente existente será apagado.

O comando deverá ser passado conforme o esquema abaixo:

<GET NAME=CAMPO1 TYPE=VALUE SIZE=X COL=Y LIN=Z NOENTER=1 ZL>

### **3.2.4 NUMERO DE SÉRIE**

Para capturar o número de série do terminal, devemos passar para o parâmetro TYPE o argumento SERIALNO.

O parâmetro NAME é sempre obrigatório.

Assim, o parâmetro NAME=xxx é sempre retornado pelo POS.

<GET TYPE=SERIALNO NAME=xxx>

### 3.2.5 CÓDIGO DE BARRAS

Para fazer leitura de dispositivo de código de barras através da entrada USB do terminal, devemos passar para o parâmetro TYPE o argumento CODEBAR.

O parâmetro NAME é sempre obrigatório.

Assim, o parâmetro NAME=xxx é sempre retornado pelo POS.

<GET TYPE=CODEBAR NAME=xxx>



### 3.2.6 PORTA SERIAL

Apenas para os modelos que tenham esta entrada.

Para capturar informações da porta serial RS232, devemos passar para o parâmetro TYPE o argumento SERIAL.

O parâmetro NAME é sempre obrigatório.

SIZE especifica o tamanho dos dados que serão lidos.

Existe um timeout que pode ser alterado na configuração, que ocorrerá caso não seja lido nada ou lido um tamanho menor que o informado.

O parâmetro NAME=xxx é sempre retornado pelo POS.

<GET TYPE=SERIAL NAME=xxx SIZE=yyy>

### 3.2.7 CONTROLES DA PORTA SERIAL

Apenas para os modelos que tenham esta entrada.

Para capturar informações da porta serial RS232, devemos passar para o parâmetro TYPE o argumento SERIALINE.

O parâmetro NAME é sempre obrigatório.

CONTROL especifica a máscara dos bits que serão lidos.

O parâmetro NAME=xxx é sempre retornado pelo POS.

<GET TYPE=SERIALINE NAME=xxx CONTROL=X>

Onde X é o valor da máscara dos bits que será enviado, conforme abaixo:

8: DCD

32: CTS

Caso queira receber os pinos DCD e CTS, usar o valor 40 (8+32).

### 3.2.8 VERSÃO

Para capturar a identificação da versão do NAVS, devemos passar para o parâmetro TYPE o argumento VERSION.

O parâmetro NAME é sempre obrigatório.

Assim, o parâmetro NAME=xxx é sempre retornado pelo POS.

<GET TYPE=VERSION NAME=xxx>

### **3.2.9 VERSÃO DO SISTEMA OPERACIONAL**

Para capturar a identificação da versão do sistema operacional do POS, devemos passar para o parâmetro TYPE o argumento OSVERSION.

O parâmetro NAME é sempre obrigatório.

Assim, o parâmetro NAME=xxx é sempre retornado pelo POS.

<GET TYPE=OSVERSION NAME=xxx>

### 3.2.10 MODELO

Para capturar o modelo do terminal POS, devemos passar para o parâmetro TYPE o argumento MODEL. O parâmetro NAME é sempre obrigatório.

Assim, o parâmetro NAME=xxx é sempre retornado pelo POS, que poderá ser “VX520” ou “VX680”

<GET TYPE=MODEL NAME=xxx>

### **3.2.11 IDENTIFICAÇÃO DO CHIP**

Para capturar a identificação do chip GPRS, devemos passar para o parâmetro TYPE o argumento ICCID. O parâmetro NAME é sempre obrigatório.

Assim, o parâmetro NAME=xxx é sempre retornado pelo POS, com a identificação do chip.

<GET TYPE=ICCID NAME=xxx>

## Especificação de Interface

---

### 3.2.12 MIFARE

Apenas para os modelos de POS que tenham esta entrada.

Para capturar os dados de identificação de cartão Mifare (contactless), devemos passar para o parâmetro TYPE o argumento MIFARE.

O comando deve ser passado conforme abaixo, notando que a ordem dos parâmetros não altera o resultado.

<GET TYPE=IDMIFARE NAME=xxxxx RC\_NAME=vvv>

vvv é o nome da variável que receberá o código de retorno do comando que deverá ser '00' se o comando foi bem sucedido.

### 3.2.13 CTLS

Apenas para os modelos de POS que tenham esta entrada.

Para capturar os dados de identificação de cartão do tipo ISSO (contactless) ou MIFARE (contactless), devemos passar para o parâmetro TYPE o argumento CTLS.

O comando deve ser passado conforme abaixo, notando que a ordem dos parâmetros não altera o resultado.

<GET TYPE=IDCTLS NAME=xxxxx>

Obs.: Os dois comandos acima (Mifare e CTLS) não podem ser usados juntos.

### 3.2.14 LEITURA DE DADOS DO MIFARE

Apenas para os modelos de POS que tenham esta entrada.

Para efetuar a leitura dos dados de um determinado setor do cartão Mifare (contactless), devemos passar para o parâmetro TYPE o argumento MIFARE\_READ.

O comando deve ser passado conforme abaixo, notando que a ordem dos parâmetros não altera o resultado.

```
<GET TYPE=MIFARE_READ NAME=xxxxx KEY=yyyyyyyyyyyyyy KTYPE=a BLOCK=01 SECTOR=0  
BCOUNT=2 RC_NAME=vvv>
```

Onde:

NAME: variável de retorno do comando.

KEY: chave para leitura do setor.

KTYPE: tipo da chave, pode ser 'a' ou 'b'.

BLOCK: bloco a ser lido.

SECTOR: setor a ser lido.

BCOUNT: contador de blocos.

Vvv é o nome da variável que receberá o código de retorno do comando que deverá ser '00' se o comando foi bem sucedido.



### **3.2.15 TOUCH (somente no Vx680)**

Para capturar informações na console do modelo Vx680, através do 'touch' na tela, devemos passar para o parâmetro TYPE o argumento TOUCH.

O comando deve ser passado conforme o esquema abaixo, notando que a ordem dos parâmetros não altera o resultado.

<GET TYPE=TOUCH NAME=xxxxx>

O comando retorna na variável NAME, o nome da figura ativa que foi tocada.

## Especificação de Interface

---

Como requisito básico do comando TOUCH, figuras devem ser desenhadas na tela, conforme comandos a seguir:

**Obs.: Considerar a tela do terminal Vx680 com: 240 x 320 pixels.**

### Retângulo:

<RECTANGLE NAME=XXX X=x Y=y WIDTH=zzz HEIGHT=www VISIBLE=k>

Onde:

X e Y são as coordenadas origem do retângulo, considerado a partir do alto a esquerda.

X é a coordenada na horizontal e Y na vertical.

WIDTH é o tamanho da largura do retângulo em pixels.

HEIGHT é o tamanho da altura do retângulo em pixels.

VISIBLE indica se a figura será visível ou não. Valores: 0 (não visível) ou 1 (visível).

### Círculo:

<CIRCLE NAME=XXX X=x Y=y RADIUS=zzz VISIBLE=k>

Onde:

X e Y são as coordenadas do centro do círculo.

X é a coordenada na horizontal e Y na vertical.

RADIUS é o tamanho do raio do círculo em pixels.

VISIBLE indica se a figura será visível ou não. Valores: 0 (não visível) ou 1 (visível).

### Polígono:

<POLYGON NAME=XXX VISIBLE=k>

X,Y;

X,Y;

X,Y;

</POLYGON>

Onde:

X e Y são as coordenadas dos vértices do polígono.

X é a coordenada na horizontal e Y na vertical.

VISIBLE indica se a figura será visível ou não. Valores: 0 (não visível) ou 1 (visível).

Exemplo para X e Y de polígono triângulo:

<POLYGON NAME=POLIG VISIBLE=1>

20,20;

20,70;

50,50;

</POLYGON>

A ordem dos vértices é importante.

OBS.: TODAS AS FIGURAS ACIMA NÃO REMOVIDAS DA MEMORIA QUANDO A TECLA 'CANCELAR' É PRESSIONADA OU QUANDO COMANDO CONSOLE É DADO.

### 3.2.16 DATA E HORA

Para capturar a data e hora do terminal, devemos passar para o parâmetro TYPE o argumento DATETIME. O parâmetro NAME é sempre obrigatório.

Assim, o parâmetro NAME=xxx é sempre retornado pelo POS.

<GET TYPE=DATETIME NAME=xxx>

A variável NAME receberá a data e hora no formato: AAAAMMDDHHMMSS.

## Especificação de Interface

---

### 3.2.17 CPF

Para capturar um CPF, por meio do teclado devemos passar para o parâmetro TYPE o argumento CPF. Neste caso haverá os parâmetros mandatórios: COL, LIN.

COL especifica a coluna da esquerda para direita onde o primeiro caractere capturado será mostrado no display.

LIN especifica a linha de cima para baixo onde o primeiro caractere capturado será mostrado no display.

NOENTER=1 especifica que o comando não aguardará o <ENTER> no final da entrada (opcional).

Devemos notar que na linha onde será mostrado o campo, todo texto previamente existente será apagado.

O comando deverá ser passado conforme o esquema abaixo:

<GET TYPE=CPF NAME=xxx COL=Y LIN=Z NOENTER=1>

O comando só retorna números válidos e com campos sem formatação. Ex.: CPF capturado = 305.123.456-78, valor de CPF retornado = 30512345678.

Obs.: O usuário não precisa inserir a formatação na tela, ela aparece conforme os campos vão sendo editados.

### 3.2.18 HOUR

Para capturar um horário, por meio do teclado devemos passar para o parâmetro TYPE o argumento HOUR.

Neste caso haverá os parâmetros mandatórios: COL, LIN.

COL especifica a coluna da esquerda para direita onde o primeiro caractere capturado será mostrado no display.

LIN especifica a linha de cima para baixo onde o primeiro caractere capturado será mostrado no display.

NOENTER=1 especifica que o comando não aguardará o <ENTER> no final da entrada (opcional).

Devemos notar que na linha onde será mostrado o campo, todo texto previamente existente será apagado.

O comando deverá ser passado conforme o esquema abaixo:

```
<GET TYPE=HOUR NAME=xxx COL=Y LIN=Z NOENTER=1>
```

O comando só retorna horários válidos e com campos sem formatação. Ex.: HOUR capturada = 22:33:13, valor de HOUR retornado = 223313.

Obs.: O usuário não precisa inserir a formatação na tela, ela aparece conforme os campos vão sendo editados.

### 3.2.19 DATA

Para capturar uma data, por meio do teclado devemos passar para o parâmetro TYPE o argumento DATA. Neste caso haverá os parâmetros mandatórios: COL, LIN, SIZE.

SIZE especifica o tamanho em caracteres do campo a ser capturado, deve ser 6 (Ex. de saída: 181013) ou 8 (Ex. de saída : 18102013)

COL especifica a coluna da esquerda para direita onde o primeiro caractere capturado será mostrado no display.

LIN especifica a linha de cima para baixo onde o primeiro caractere capturado será mostrado no display.

NOENTER=1 especifica que o comando não aguardará o <ENTER> no final da entrada (opcional).

Devemos notar que na linha onde será mostrado o campo, todo texto previamente existente será apagado.

O comando deverá ser passado conforme o esquema abaixo:

<GET TYPE=DATA NAME=xxx SIZE=8 COL=Y LIN=Z NOENTER=1>

O comando só retorna datas válidas e com campos sem formatação. Ex.: DATA capturada = 18/10/2013, valor de DATA retornado = 18102013.

Obs.: O usuário não precisa inserir a formatação na tela, ela aparece conforme os campos vão sendo editados.

### 3.2.20 CNPJ

Para capturar um CNPJ, por meio do teclado devemos passar para o parâmetro TYPE o argumento CNPJ. Neste caso haverá os parâmetros mandatórios: COL, LIN.

COL especifica a coluna da esquerda para direita onde o primeiro caractere capturado será mostrado no display.

LIN especifica a linha de cima para baixo onde o primeiro caractere capturado será mostrado no display.

NOENTER=1 especifica que o comando não aguardará o <ENTER> no final da entrada (opcional).

Devemos notar que na linha onde será mostrado o campo, todo texto previamente existente será apagado.

O comando deverá ser passado conforme o esquema abaixo:

<GET TYPE=CNPJ NAME=xxx COL=Y LIN=Z NOENTER=1>

O comando só retorna números válidos e com campos sem formatação. Ex.: CNPJ capturado = 11.444.777/0001-61, valor de CNPJ retornado = 11444777000161.

Obs.: O usuário não precisa inserir a formatação na tela, ela aparece conforme os campos vão sendo editados.

### 3.2.21 APDUCHIP

<GET TYPE=APDUCHIP NAME=RESP APDU=xxxxxx SLOT=1 ON=N OFF=N RC\_NAME =rrr ATR\_NAME =aaa>

Onde rrr é o nome da variável que receberá o return-code  
E aaa é o nome da variável que receberá o ATR no caso de ON=Y

NAME é o campo que recebe o retorno da chamada da apdu  
SLOT =0, 1, 2 ou 3, sendo 0 o cartão externo e os demais os SAMs  
ON=Y se quer que ligue o cartão, ON=N caso contrário  
OFF=Y se quer que desligue o cartão, OFF=N caso contrário  
APDU é a apdu em hexa

O comando ainda retorna RC\_NAME que deve ser 00 se tudo for ok  
ATR\_NAME com o valor do ATR em hexa  
APDU com o valor da apdu enviada

OBS: RC\_NAME =00 não significa que a apdu funcionou, e sim que o comando funcionou. A resposta da APDU deve ser verificada  
Caso RC\_NAME não seja 00, o campo indicado por NAME tem o erro.

O limite de bytes hexa a ser enviado para APDUs é de 150 bytes o que dá uma APDU binária de 75 bytes. Caso seja necessário uma APDU maior, usar os parâmetros APDU1, APDU2, APDU3, APDU4 e APDU5 respeitando essa ordem. Os conteúdos serão concatenados ao parâmetro APDU. Lembramos que, para cada parâmetro APDUx, o limite também é de 150 bytes hexa. Só usar os parâmetros necessários para conseguir enviar a totalidade de bytes desejada.



### 3.2.22 DESNBS

<GET TYPE=DESNBS CODOP=x KEY=kkkk IN=yyyy NAME=nnnn RC\_NAME=rrr >

Criptografia no padrão DES.

X = 0 cifra

X = 1 decifra

KEY = chave de criptografia com 8 bytes em hexa (16 caracteres)

IN = campo a (des)criptografar com 8 bytes em hexa (16 caracteres)

NAME = nome do campo (des)criptografado com 8 bytes em hexa (16 caracteres)

O comando ainda retorna RC\_NAME que deve ser 00 se tudo for ok

Caso RC\_NAME não seja 00, o campo indicado por NAME tem o erro.

## Especificação de Interface

---

### 3.2.23 INFO\_FILE

<GET TYPE=INFO\_FILE NAME=xxx ID\_FILE=yyyy>

Comando que retorna, se o arquivo existir no POS, tamanho, data e hora da criação ou data e hora da última alteração do arquivo.

Onde :

Xxx é o nome da variável que recebera o retorno.

Yyy é o nome do arquivo com sua respectiva extensão (Exs.: imagem.bmp, contabilidade.lua, etc..)

Os retornos esperados são:

“ERRO” em caso de o arquivo não existir.

“TAMANHO\_DATA\_HORA”, caso o arquivo exista. (Exemplo : 2112\_17052013\_192522, onde 2112 é o tamanho em bytes do arquivo, 17/05/2013 é a data de criação ou última data de modificação do arquivo, e 19:25:22 é o horário de criação ou última modificação do arquivo).

DATA vem no formato Dia/Mês/Ano.

E HORA vem no formato Hora:Minuto:Segundo.

## Especificação de Interface

---

### 3.2.24 DRIVE\_SIZE

<GET TYPE=DRIVE\_SIZE NAME=xxxxx ID\_DRIVE=yyyyy>

Comando que retorna, o tamanho total do drive do POS (memória), tamanho disponível (em Kb) e quantidade de arquivos no drive.

Onde :

Xxxxx é o nome da variável que receberá o retorno.

Yyyyy é a identificação do drive:

“I.” – RAM

“F.” – FLASH

Os retornos esperados são:

“T=aaaa;U=bbbb;Q=cccc”, string com os dados do drive.

T, informa o Total da Memória do drive especificado.

U, informa o tamanho da memória em uso atualmente, do drive especificado.

Q, informa a quantidade de arquivos do drive especificado.

Se o conteúdo deles for “0”, indica que o drive não existe.

### 3.2.25 EMV

<GET TYPE=EMV NAME=RESP RC\_NAME=rrrr REDEADQ=x AID=yyyyyyy TIPOAPP=z>

Leitura de cartão EMV, via chip ou sem contato (ctls), mas também efetua leitura de tarja.

RC\_NAME = Nome da variável que receberá o return-code.

REDEADQ = Rede adquirente: 1 (Amex), 2 (Rede), 3 (Cielo).

AID = Identificador da aplicação, exemplo: A00000000031010. (até 32 dígitos)

TIPOAPP = Tipo de aplicação: 1 (Crédito), 2 (Débito), 3 (Moedeiro).

O comando ainda retorna RC\_NAME que deve ser 00 se tudo for ok

Caso RC\_NAME não seja 00, o campo indicado por NAME tem o erro.

Abaixo os erros mais comuns retornados na variável de RC\_NAME:

12 – Esgotado o tempo máximo estipulado para a operação.

13 – Operação cancelada pelo operador.

22 – Aplicação da rede adquirente não existente.

41 – Erro de leitura do cartão magnético.

60 – Cartão não responde (“mudo”) ou chip não presente.

61 – Erro de comunicação do POS com o cartão inteligente.

62 – Cartão do tipo inválido ou desconhecido, não pode ser tratado (não é EMV).

66 – Cartão com erro de estrutura – arquivos estão faltando.

67 – Cartão foi invalidado.

68 – Cartão com problemas. Esse status é válido para muitas ocorrências no processamento de cartões EMV, onde o cartão não se comporta conforme o esperado e a transação deve ser finalizada.

69 – Cartão possui dados inválidos ou inconsistentes.

70 – Cartão sem nenhuma aplicação disponível para as condições pedidas ou cartão é reconhecido como EMV mas não possui nenhuma aplicação compatível com a requerida.

75 – Cartão possui somente uma aplicação candidata e ela está bloqueada.

76 – Erro de alto nível no cartão EMV que é passível de “fallback” para tarja magnética.

### 3.3 *DELAY*

Usar este comando quando quiser que o POS aguarde algum tempo antes de continuar qualquer outro comando.

Ideal para mensagens para a Console temporizadas.

<DELAY TIME=xx>

Onde xx é o tempo de espera em segundos.

Exemplo para uso com mensagem para console temporizada em 3 segundos:

<CONSOLE> Texto para display </CONSOLE>

<DELAY TIME=3>

### 3.4 **SELECT**

Para selecionar uma opção a partir de um menu de itens de caracteres, utilizar o comando abaixo:

```
<SELECT LIN=3 COL=4 SIZE=15 QTD=3 UP=E1 DOWN=E2 RIGHT=B7 LEFT=B9 NAME=SEL
TYPE_RETURN=3 INDEX=n1,n2,n3,...,nX>
item1,
item2,
item4,
etc,
fim
</SELECT>
```

Onde,

LIN = linha inicial da lista

COL=coluna inicial da lista

SIZE=Tamanho de cada item da lista (o que será mostrado em reverso)

QTD=Quantidade de itens da lista a aparecerem da tela

UP=tecla de paginação para cima

DOWN= tecla de paginação para baixo

NAME=nome da variável que retornara o valor selecionado pelo ENTER da lista

“item1,item2,item4,etc,fim”= exemplo de lista de valores entre vírgulas.

RIGHT= tecla para fazer a rolagem da string na horizontal, para o lado direito, caso ela seja maior que área de apresentação.

LEFT= tecla para fazer a rolagem da string na horizontal, para o lado esquerdo, caso ela seja maior que área de apresentação.

TYPE\_RETURN é o parâmetro que define se o retorno será a string selecionada ou o número associado à ela na ordem em que as strings são listadas. Assume os valores 1 (retorna string), ou 2(retorna o número correspondente), ou 3(retorna o índice aleatório correspondente).

INDEX é o parâmetro que recebe os valores dos índices aleatórios desejados. Note que n1 a nX devem ser números INTEIROS que são separados por vírgula. **O finalizador de INDEX também é a vírgula.**

Para inserir espaços usar %20:

Exemplo: “não 54erfo”, tem que ficar: “não%20paulo”.

Para inserir “>” e “<” usar %3E e %3C respectivamente:

Exemplo1: “1-numeromax>500”, tem que ficar: “1-numeromax%3E500”.

Exemplo2: “2-numeromax<250”, tem que ficar: “2-numeromax%3C250”.

Para inserir “,” usar %2C:

Exemplo: “0,5”, tem que ficar: “0%2C5”.

Obs.: Este comando está relacionado a “<CONSOLE>xx</CONSOLE>”, porém deverá ser acionado fora dele.

Abaixo alguns exemplos de caracteres hexadecimal das teclas para o SELECT:

TECLA	VALOR HEXA
F1	FA
F2	FB
F3	FC
F4	FD
Roxa 1	E1
Roxa 2	E2
Roxa 3	E3
Roxa 4	E4
*	AA
#	A3
0	B0
1	B1
2	B2
3	B3
4	B4
5	B5
6	B6
7	B7
8	B8
9	B9

### 3.5 *FTP*

Comando para fazer download ou upload de arquivo via FTP.

Exemplo de uso:

```
<FTP TYPE=a HOST=b PORT=c USERID=d PASSW=e LOCALFILE=f REMOTEFILE=g RETURN=H
MODE=PASSIVE HOSTFTP=i PORTFTP=j>
```

Onde :

“a” pode ser GET(para fazer download) ou PUT(para fazer upload).

“b” é o host de destino, pode ser DNS ou IP.

“c” é a porta FTP do host destino.

“d” é a identificação do usuário.

“e” é a senha do usuário.

“f” é o nome do arquivo que está ou virá para o POS, no Verifone sempre deverá vir com “I:” na frente do nome do arquivo. (Exemplo: “I:nomeArquivo.extensão”).

“g” é o nome do arquivo que está ou irá para o servidor.

E “H” é a variável que irá receber o retorno de status da operação. Os valores retornados serão 0 (sucesso na operação), 1 (erro), e 2 (sem conexão).

MODE=PASSIVE deve ser usado no uso de FTP passivo.

“i” é o IP ou DNS do host ftp no modo passivo (algumas operadoras de chip exigem este campo)

“j” é a porta host ftp no modo passivo (algumas operadoras de chip exigem este campo)

Abaixo alguns códigos de erro que podem ocorrer no uso do modo FTP PASSIVO:

FTP\_Cmd\_GetFile:

```
3;      // Erro ao criar arquivo local
4;      // Erro ao escrever no arquivo local
```

FTP\_Cmd\_TYPE\_I:

```
5 e 6;  // Erro no Parser do comando TYPE
```

FTP\_Cmd\_PASS:

```
7, 8;   // Erro no Parser do comando PASS
9;      // Erro comando PASS rejeitado
10;     // Erro no Parser do comando PASS
```

FTP\_Cmd\_USER:

```
11, 12, 13; // Erro no Parser do comando USER
14 e 15;    // Erro comando USER rejeitado
16;        // Erro no Parser do comando USER
```



## Especificação de Interface

---

### FTP\_Cmd\_PASV:

17, 18, 19, 20, 21, 22, 23, 24; // Erro no Parser do comando PASV  
25; // Falha no HOST comando TCP\_connectData

### FTP\_Cmd\_RETR:

26, 27, 28; // Erro no Parser do comando RETR  
29; // Sem conexão TCP  
30; // Erro Leitura Arquivo Remoto  
31; // Sem conexão TCP  
32; // Erro Leitura Arquivo Remoto  
33; // Erro no Parser do comando RETR

### FTP\_connect:

34; // Falha no HOST 57ommando TCP\_connect  
47; // Falha no DNS

### FTP\_Cmd\_PutFile:

35; // Erro ao abrir arquivo local  
36; // Erro ao ler o arquivo local  
37; // Erro ao enviar o arquivo local  
38; // Enviou tam diferente do lido

### FTP\_Cmd\_STOR:

39, 40, 41; // Erro no Parser do comando STOR  
42; // Sem conexão TCP  
43; // Erro Leitura Arquivo Remoto  
44; // Sem conexão TCP  
45; // Erro Leitura Arquivo Remoto  
46; // Erro no Parser do comando STOR

### 3.6 FINGERPRINT

Comando para captura de dados através da biometria, somente para os modelos que possuem adaptação para conectar o leitor biométrico (caso Verifone) ou possuem o dispositivo integrado (caso Ingenico).

O uso da biometria tem 3 partes distintas:

1 – Capturar o padrão do fingerprint para efeitos de cadastro para futuras comparações:

```
<FINGERPRINT NAME=nnnn TYPE=CAPTURE USE=REGISTER FORMAT=fff VERBOSE=Y LIN=5
COL=1 RET=rrr></FINGERPRINT>
```

58ff ff pode ser ISSO ou RAW;

Verbose indica se as mensagens de captura aparecem na tela, neste caso LIN e COL indica em qual posição;

rrr – o nome de um variável que terá o retorno do comando (se ok ou não);

nnnn – o nome da variável com os dados do dedo capturado para armazenar em seu banco de dados;

1 – Capturar o dedo para comparação:

```
<FINGERPRINT NAME=nnnn TYPE=CAPTURE USE=VERIFY FORMAT=fff VERBOSE=Y LIN=5 COL=1
RET=rrr></FINGERPRINT>
```

A diferença é que a variável indicada em nnnn possuirá o fingerprint para bater com os cadastrados em seu banco de dados.

1 – Aqui é a fase de checar o que foi capturado na parte 2 com o armazenado previamente na parte 1. Isso pode ser feito por rotinas no seu servidor (para tal usar o formato ISSO nas partes 1 e 2). Caso deva ser feito no seu POS, usar o formato RAW nas partes 1 e 2 e usar o comando abaixo para checar:

```
“<FINGERPRINT NAME=nnnn TYPE=VERIFY FORMAT=RAW MATCH=mmmm RET=rrr>
VERIFY=vvvv REGISTER=rrrr
</FINGERPRINT>
```

Onde mmmm é a variável que possuirá o retorno da verificação

vvvv é o conteúdo capturado na parte 2 pela variável nnnn

rrrr é o conteúdo capturado na parte 1 pela variável nnnn

Obs: essa parte é a mais demorada pois o comando deve ser repetido para cada dedo capturado e armazenado nas bases de dados.

Sintaxe:

## Especificação de Interface

---

<FINGERPRINT NAME=xxx TYPE=init RET=rrr></FINGERPRINT>

<FINGERPRINT NAME=xxx TYPE=capture USE=verify FORMAT=raw RET=rrr ></FINGERPRINT>

<FINGERPRINT NAME=xxx TYPE=capture USE=register FORMAT=raw RET=rrr ></FINGERPRINT>

<FINGERPRINT NAME=xxx TYPE=verify FORMAT=raw MATCH=yyy RET=rrr >  
VERIFY=xxxxxxxxxxxxx REGISTER=xxxxxxxxxxxxxxxxx  
</FINGERPRINT>

Importante: O TYPE=init deve ser colocado ANTES do primeiro <INIT>.

O USE=register serve pra capturar o dedo para cadastro.

O USE=verify serve pra capturar o dedo para checar.

No TYPE=verify, o VERIFY=xxxxx tem o conteúdo devolvido no TYPE=capture USE=verify e o REGISTER=xxxxx tem o conteúdo devolvido no TYPE=capture USE=register.

O MATCH=yyy indica a variável que terá o resultado da verificação (0 = não bate – 1 = bate).

FORMAT recebe o valor “raw”(captura a imagem e armazena no formato padrão).

NAME é a variável que recebe o retorno da operação.

RET é o nome da variável que recebe o retorno do comando FINGERPRINT.

Um Beep na leitura é gerado.

### **3.7 TONE**

Usar este comando quando quiser fazer com que o POS emita sons.

Exemplo de uso:

<TONE TYPE=xxx>, onde “xxx” pode ser:

NORMAL: emite um beep indicando normalidade.

ERROR: emite um beep indicando anormalidade.

### 3.8 **PRINTER**

<PRINTER PRNSTS=STS ENABLE\_CANCEL=xxx> inicia texto para a impressora.

PRNSTS informa a variável que terá o retorno do status final da operação de impressão,

</PRINTER> finaliza texto para a impressora,

A mudança de linha é realizada pelo comando <BR>.

ENABLE\_CANCEL, especifica se tecla CANCEL será ativa ou não após o acionamento da impressão, até o envio do comando <POST>.

Onde xxx, recebe os seguintes valores:

0 = permite cancelamento;

1 = não cancela nada até os dados serem enviados ao servidor (comando POST).

Obs.: O texto máximo para impressora é de 3.000 caracteres.

PRNSTS, retorna na variável (Ex.: STS, o status da impressão)

Os valores de status retornados são:

0: Ok;

-4: Falha;

-5: Pouco papel;

-10: erro de RAM;

-20: Falha na impressora;

-21: Sem papel;

-23: Sequência de Escape Code não encontrada;

-24: Impressora não inicializada;

-27: Firmware corrompido.

Obs.2: Qualquer outro comando deverá ser utilizado fora de "<PRINTER PRNSTS=STS >xx</ PRINTER >", entre eles haverá SOMENTE texto para o display.

### 3.9 **CANCEL\_KEY**

Desabilita ou habilita a tecla Cancel (X).

<CANCEL\_KEY TYPE=xxxxxx> onde xxxxxx pode ser:

DISABLE: desabilita a tecla Cancel, no comando POST a tecla Cancel volta a ficar habilitada automaticamente.

ENABLE: habilita a tecla Cancel.

### 3.10 *CHGCONFNT*

Comando que permite a mudança de fontes para o console.

<CHGCONFNT IDFONT=MODEL1 SIZE=10 INVERSE UNDERLINE>, onde :

IDFONT, indica a fonte a ser utilizada, assumindo os seguintes valores:

DEFAULT, onde o modelo padrão do POS é selecionado, seu tamanho será sempre 8.

MODEL1, corresponde ao fonte Arial, pode assumir os tamanhos 8 e 12.

MODEL2, corresponde ao fonte ASC, pode assumir o tamanho 16, suportado apenas no modelo Vx520.

MODEL3, corresponde ao fonte Courier New, pode assumir os tamanhos 12 e 16.

MODEL4, corresponde ao fonte Helvética, pode assumir o tamanho 16, suportado apenas no modelo Vx520.

MODEL5, corresponde ao fonte K2, pode assumir o tamanho 16.

MODEL6, corresponde ao fonte Times New Roman, pode assumir os tamanhos 12 e 16.

MODEL7, corresponde ao fonte Verdana, pode assumir os tamanhos 12 e 16.

SIZE, indica o tamanho do fonte selecionado.

INVERSE e UNDERLINE são parâmetros opcionais, que podem ser aplicados em todos os fontes exceto o DEFAULT.

### 3.11 *PRNFNT*

Para imprimir texto com fontes diferentes usar o comando abaixo:

<PRNFNT SIZE=x PARAMETROS>

Onde,

X = tamanho da fonte, no range de 2 a 4 (4 fonte menor, e 2 fonte maior).

2: 5x8 24 colunas.

3: 8x14 32 colunas.

4: 8x14 42 colunas.

*OBS.: o size 4 é o padrão inicial do NAVS.*

PARAMETROS:

DBL\_HEIGHT: dobro da altura

DBL\_WIDTH: dobro da largura

INVERSE: inverte as cores

Os parâmetros DBL\_HEIGHT, DBL\_WIDTH, e INVERSE, podem ser utilizados juntos no mesmo comando, com espaço entre eles.

Obs2.: Este comando está relacionado a "<PRINTER PRNSTS=STS >xx</ PRINTER >", porém deverá ser acionado fora dele.

### 3.12 *WRITE\_AT*

<WRITE\_AT LINE=1 COLUMN=1> inicia texto para o display,  
</WRITE\_AT> finaliza texto para o display.

Não realiza mudança de linha.

Sobrescreve qualquer coisa que estiver na tela, sem apagar o conteúdo total dela, tendo início na linha e coluna definidos respectivamente pelos parâmetros LINE e COLUMN.

Obs.: Qualquer outro comando deverá ser utilizado fora de  
“<WRITE\_AT LINE=x COLUMN=y> ... </WRITE\_AT>”, entre eles haverá SOMENTE texto para o display.

Obs.: A partir da versão 1.17.0, escreve apenas nos espaços do texto escrito (não apaga mais a linha toda).



### 3.13 *DRAW (Somente para Vx680)*

Para capturar um desenho feito na tela do VX680 e a partir dele gerar um arquivo com extensão “.bmp” monocromático, utilize o comando abaixo:

```
<DRAW BMP=desenho.bmp TRACE=2>
```

Onde:

BMP é o parâmetro que recebe o nome do arquivo bmp a ser gerado, que deverá vir sempre no seguinte formato “nome\_desejado”+”.bmp”.

TRACE é o parâmetro que define a intensidade do traçado na tela, com valores a partir de 1.

Para finalizar a captura de traçado na tela deve-se utilizar o comando abaixo:

```
</DRAW>
```

Dentro do seguinte conjunto:

```
<DRAW BMP=desenho.bmp TRACE=2>  
</DRAW>
```

Poderá utilizar outros comandos como por exemplo :

- <GET TYPE=ANYKEY> , o que fará com que o usuário desenhe, e após concluir possa pressionar a tecla VERDE e somente depois disso gerar o arquivo bmp.
- <CAPTURE NAME=xxx>

```
<GET TYPE=FIELD ALPHA=1 SIZE=X COL=Y LIN=Z NOENTER=1>  
<GET TYPE=SPECIALKEY>  
</CAPTURE>
```

### 3.14 CHGPRNFNT

Para imprimir texto com fontes diferentes usar o comando abaixo:

<CHGPRNFNT SIZE=X FACE=FACE1 PARAMETRO1 PARAMETRO2>

Onde,

X = tamanho da fonte, no range de 2 a 4 (4 fonte menor, e 2 fonte maior).

2: 5x8 24 colunas.

3: 8x14 32 colunas.

4: 8x14 42 colunas.

*OBS.: o tamanho 4 é o padrão inicial do NAVS.*

FACE = identificação da fonte:

“FONTE1” é modelo default do POS.

“FONTE2”.

“FONTE3”.

“FONTE4”.

“FONTE5”.

“FONTE6”.

Obs.: “FONTE1” e “FONTE2” não suportam os parâmetros “UNDERLINE” e “BOLD”.

Obs2.: “FONTE3” e “FONTE4” não suportam o tamanho 2, somente 3 e 4.

Obs3.: “FONTE5” e “FONTE6” com os parâmetros “INVERSE”, “BOLD”, “UNDERLINE” juntos não suportam o tamanho 4, somente 2 e 3.

PARAMETROS:

DBL\_HEIGHT: dobro da altura

DBL\_WIDTH: dobro da largura

BOLD: formato negrito.

INVERSE: inverte os bits da fonte.

UNDERLINE: sublinha os caracteres.

Os parâmetros: “DBL\_HEIGHT”, “DBL\_WIDTH”, “BOLD”, “INVERSE”, e “UNDERLINE”, podem ser utilizados juntos no mesmo comando, com espaço entre eles.

Obs4.: Este comando está relacionado a “<PRINTER PRNSTS=STS >xx</ PRINTER >”, porém deverá ser acionado fora dele.

Obs5.: Todos os parâmetros devem ser passados com letras maiúsculas.

A seguir , os exemplos dos modelos de todos os fontes disponíveis:

[illegible]

### 3.15 PRNLOGO

Para imprimir arquivo Imagem usar o comando abaixo:

<PRNLOGO NAME=xxxx.bmp SPACES=Y>

Onde,

xxxx.bmp (name) é o nome do arquivo a ser impresso (sempre em minúsculo, e com no máximo 29 caracteres). O arquivo já deve estar carregado no POS.

Y (spaces) são os espaços à esquerda para centralização.

A imagem terá que ser no formato bmp e monocromática.

O tamanho máximo da largura da imagem é de 384 pixels.

Não há limite para o comprimento.

### 3.16 CONLOGO

Para mostrar no display do POS um arquivo Imagem usar o comando abaixo:

```
<CONLOGO NOCLS=pppp NAME=xxxx.bmp X=x Y=y>
```

Onde,

xxxx.bmp (name) é o nome do arquivo que será colocado na tela do POS (sempre em minúsculo). O arquivo já deve estar carregado no POS.

X e Y são as coordenadas do display do POS.

“pppp” é o valor de NOCLS.

O parâmetro NOCLS determina se uma imagem que estiver no display será apagada ou não com a atualização do texto na tela.

Os valores que podem ser assumidos são 0=não 1=sim.

Para o modelo Vx520 o arquivo tem que ser do tipo bmp e monocromático. Tamanho máximo (128x64).

Já para o Vx680 tem que ser também no formato bmp, porém pode ser colorido. Se a imagem tiver exatamente 240x320 (dimensões máximas do POS), terá que ser salva com intensidade de 8 bits, se for menor que a resolução máxima (240x320) a intensidade de bits poderá ser maior.

Para o modelo E265, o arquivo tem tamanho máximo de 156x110 monocromático.

### 3.17 ANIMATE

Comando para exibir imagens animadas (por exemplo, um arquivo GIF).

<ANIMATE FILENAME=imagem.gif NUM\_FRAMES=12 ROW=100 COLUMN=100>

O parâmetro **FILENAME** especifica o nome do arquivo que será exibido na tela do POS. O arquivo já deve estar carregado no POS.

O parâmetro **ROW** e **COLUMN** são as coordenadas do display do POS. Observe que as coordenadas são medidas em PIXELS.

O parâmetro **NUM\_FRAMES** especifica o número de quadros da animação. Se o número de quadros do arquivo GIF for menor que NUM\_FRAMES, então será respeitado o número de quadros do arquivo GIF.

#### Observações importantes deste comando:

Tamanho máximo da imagem: 32 x 32 pixels

Profundidade de cores: **somente** 8 bits por pixel / 256 cores

Número máximo de quadros para a animação: 12 quadros

É recomendável utilizar o editor MS Paint (editor do próprio Windows) para criar as imagens GIF.

Os parâmetros ROW e COLUMN especificam o posicionamento (em pixels) da imagem na tela do POS. A origem (0, 0) fica no canto superior esquerdo.

### 3.18 CODEBAR – Impressão

<CODEBAR VALUE=XXXXXXXXXXXX SIZE=Y HEIGHT=ZZ SPACES=W>

Onde XXXXXXXX (value), é o texto a ser convertido para o código de barras.

Y (size) é o tamanho, de 1 a 4, como recomendação usar 2.

ZZ (height) é a altura do código de barras em pixels, recomendação usar 128.

W (spaces) são os espaços à esquerda para centralização, recomendação usar 90.

O padrão utilizado é o EAN-13, onde o valor máximo para o VALUE deve ser 12. O NAVS calcula o dígito.

No EAN-13 o símbolo codifica treze números que estão divididos em quatro partes; dos treze dígitos, doze são dos dados referentes ao produto e um é o dígito verificador.

### 3.19 GERADOR DE QR CODES

Para criar um QrCode a partir de uma string usar o comando abaixo:

```
<GENERATE_QR_CODE SIZE=6 QR_ECLEVEL=3 KEEP_FILE=1 SPACES=0  
FILE_NAME=testeQRA.bmp NO_PRINT=1 ERR_QR=VAR_R>  
    TEXTO A SER CONVERTIDO EM QR CODE  
</GENERATE_QR_CODE>
```

Onde:

O texto a ser convertido em QR Code deve respeitar um limite máximo de 199 caracteres.

SIZE é o parâmetro que controla a intensidade do traço do QR Code. Se o conteúdo a ser convertido estiver próximo do limite máximo de caracteres, SIZE deve ser reduzido para se adequar às condições mínimas de geração do QR Code. Se o conteúdo a ser convertido contiver poucos caracteres, SIZE deve ser aumentado. Valores entre 3 e 10 são coerentes.

QR\_ECLEVEL é o parâmetro que controla a resolução do QR Code criado, o que também pode interferir no tamanho final dele. Deverá receber os seguintes valores 0 (menor resolução possível), 1, 2 ou 3 (maior resolução possível).

KEEP\_FILE é o parâmetro que indica se o arquivo QR Code criado deverá ser ou não mantido para uso posterior (Ex.: Mostrar na tela utilizando o comando CONLOGO), deverá receber os valores 0 (não manter), 1 (manter salvando invertido – para possível impressão), 2 (manter salvando com imagem original – para o display).

SPACES é o parâmetro que define o número de espaços à esquerda para a impressão do QR Code.

FILE\_NAME é o parâmetro que deverá conter o nome do arquivo, caso queira mantê-lo, a extensão dele deverá ser sempre bmp, pois o POS não trabalha com outros formatos. Todos os nomes deverão ter o seguinte formato : “nome.bmp”.

NO\_PRINT, com valores: 0 (Imprime), 1 (Não Imprime).

ERR\_QR é o parâmetro que retorna se o QR Code foi o não gerado com sucesso. Os valores retornados são 1 (representa sucesso) ou 0 (representa falha). As falhas de criação estão associadas ao tamanho pequeno do QR Code final, para corrigir isso é recomendado aumentar os valores de SIZE e QR\_ECLEVEL.

</GENERATE\_QR\_CODE> indica o encerramento do texto para ser convertido em QR Code.

Exemplo de QR Code gerado:





### 3.20 Criar Código de barras 3 de 9

Para gerar e imprimir um código de barras 3 de 9, utilizar o comando abaixo:

```
<CODE_BAR_3_9 WIDTH=130 HEIGHT =6 KEEP_FILE=0 FILE_NAME=teste3x9.bmp SPACES=0  
ERROR=RETORNO3x9>
```

**TEXTO A SER CONVERTIDO EM CODE BAR 3 DE 9**

```
</CODE_BAR_3_9>
```

Onde:

WIDTH é o parâmetro que controla o tamanho da largura do código de barras.

HEIGHT é o parâmetro que controla o tamanho da altura do código de barras.

SPACES é o parâmetro que define o número de espaços á esquerda para a impressão do Código de barras.

ERROR é o parâmetro que retorna se o Código de barras foi o não gerado com sucesso. Os valores retornados são 1 (representa sucesso) ou 0 (representa falha). As falhas de criação estão associadas ao tamanho do código de barras final (grande ou pequeno demais), para corrigir isso é recomendado tomar cuidado com o valores de WIDTH e HEIGHT. Obs.: O valor de WIDTH deverá ser superior ao valor de HEIGHT.

</CODE\_BAR\_3\_9> indica o encerramento do texto para ser convertido em Código de barras.

Obs.: A impressão do código é feita no sentido vertical.

### 3.21 Criar Código de Barras ITF (2 de 5)

Para gerar e imprimir um código de barras ITF, utilizar o comando abaixo:

```
<CODE_BAR_2_5 VALUE=0123456789 WIDTH =2 HEIGHT=120 SPACES=0 ERROR=MY_RET>
```

Onde:

VALUE é o parâmetro que recebe o valor a ser codificado para o código de barras.

WIDTH é o parâmetro que controla o tamanho da largura do código de barras.

HEIGHT é o parâmetro que recebe o tamanho da altura do código de barras.

SPACES é o parâmetro que define o número de espaços á esquerda para a impressão do código de barras.

ERROR é o parâmetro que retorna se o Código de barras foi o não gerado com sucesso. Os valores retornados são 1 (representa sucesso) ou 0 (representa falha). As falhas de criação estão associadas ao tamanho pequeno do Código de barras final, para corrigir isso é recomendado aumentar os valores de WIDTH e HEIGHT.

Obs.: A impressão do código é feita no sentido horizontal.

### 3.22 Criar Código de Barras CODE-128

O comando **CODE128** imprime o código de barras a partir do código Code-128.

Sintaxe:

<CODE128

TYPE=[A, B, C]

WIDTH=[1~4]

HEIGHT=[64~256]

SPACES=[0~90]

ORIENTATION=[HORIZONTAL, VERTICAL]

ERROR=[VARIÁVEL]>

CODIGO

</CODE128>

O parâmetro **TYPE** especifica o tipo do Code-128 que pode ser A, B ou C.

- O tipo “A” considera números, letras maiúsculas, e caracteres de controle;
- O tipo “B” considera números, letras maiúsculas e minúsculas, e caracteres adicionais;
- O tipo “C” considera apenas números.

O parâmetro **WIDTH** especifica a largura (em pixels) de cada barra ou espaço. O valor deve ser de 1 a 4.

O parâmetro **HEIGHT** especifica a altura (em pixels) da imagem. O valor deve ser de 64 a 256.

O parâmetro **SPACES** especifica o espaço horizontal para alinhar a imagem. O valor deve ser de 0 a 90.

O parâmetro **ORIENTATION** especifica a orientação da imagem que pode ser horizontal ou vertical.

O parâmetro **ERROR** retorna o código de erro da operação.

Os valores que podem ser retornados pela variável **ERROR** são:

0 – (Sucesso na operação)

1 – (Erro na operação)

– (Tipo inicial inválido para o parâmetro **TYPE** – deve ser A, B ou C)

– (Código com dígitos inválidos para Code 128 A)

4 – (Código com dígitos inválidos para Code 128 B)

5 – (Código com dígitos inválidos ou com um número ímpar de dígitos para Code 128 C)

Observações:

Os parâmetros **WIDTH**, **HEIGHT**, **SPACES** e **ORIENTATION** não são obrigatórios. Eles assumem os valores padrão 2, 128, 0 e **HORIZONTAL**, respectivamente, se não forem especificados.

Exemplo:

```
<CODE128 TYPE=C WIDTH=2 HEIGHT=128 SPACES=0  
ORIENTATION=HORIZONTAL ERROR=ERROR_RET>0123456789</CODE128>
```

### 3.23 SAÍDA SERIAL

Apenas para os modelos que tenham esta saída.

<SERIAL> inicia texto para a serial,  
</SERIAL > finaliza texto para a serial,

Obs.: O tamanho máximo para a serial é de 3.000 caracteres.

Expo: caso queira enviar: "TESTE", o comando será:  
<SERIAL>5445535445</SERIAL>, pois 'T' é '54', 'E' é '45' em ascii e assim por diante.

### 3.24 SET TIME

Comando para alterar a data e/ou hora do POS:

<SET TYPE=TIME HOUR=223400 DATE=16102002 HDSTS=STSSTT>

Onde:

HOUR: É a hora que atribuiremos ao POS, se for "0" será mantida a hora atual. Deverá estar no formato hhmmss (hh = hora, mm= minuto, ss = segundos);

DATE: É a data que atribuiremos ao POS, se for "0" será mantida a data atual. Deverá estar no formato ddmmaaaa (dd = dia, mm= mês, aaaa = ano (exemplo:2013));

HDSTS: é o retorno do status da operação, que pode ser:

- OK;
- ERRO\_LEITURA, é erro de leitura de dados do POS quando um dos parâmetros (HOUR ou DATE) vai zerado;
- ERRO\_ENTRADA, é erro no formato enviado para o POS em HOUR ou DATE;
- e ERRO\_GRAVAÇÃO, quando o POS não conseguiu efetuar a gravação dos novos parâmetros, portanto a operação não foi executada;

OBS.: As datas devem ser a partir do ano 2000, caso contrário é provável que ocorra "ERRO\_GRAVAÇÃO".

### 3.25 *SET BACKLIGHT*

Comando para alterar a intensidade da iluminação de fundo da tela LCD:

<SET TYPE=BACKLIGHT LEVEL=n>

Onde:

n=nível de intensidade de 0 a 100 sendo 0=luz desligada e 100=intensidade máxima

### **3.26 SET KBBACKLIGHT**

Comando para alterar a intensidade da iluminação de fundo do teclado:

<SET TYPE=KBBACKLIGHT LEVEL=n>

Onde:

n=nível de intensidade de 0 a 100 sendo 0=luz desligada e 100=intensidade máxima

### 3.27 KEYBOARD

Comando para captura de strings no formato alfanumérico através de um teclado touch (letras, a-z, e caracteres especiais) e teclado do POS (números), funciona somente para Vx680 e Vx685.

<KEYBOARD MODEL=A0 BEEP=ON SIZE=15 HIDE=OFF ALIGNMENT=ppp RETURN=TEC>

Onde :

MODEL = é o tipo de teclado a ser utilizado, no momento temos apenas o A0, mas posteriormente existirão outros modelos.

BEEP = define se o touch emitirá ou não som ao ser pressionado, assume valores ON (com beep) e OFF (sem beep).

SIZE = define o tamanho da string a ser capturada.

HIDE = define se a string será ou não mostrada na tela.

ALIGNMENT = define o alinhamento do texto na tela, recebe os seguintes valores : "RIGHT" (alinhamento para a direita), "LEFT" (alinhamento para a esquerda), e "CENTER" (centralizar), na ausência desse parâmetro, será considerada o alinhamento à esquerda.

RETURN = recebe a variável para retorno.

Obs.: A digitação está habilitada a partir da segunda linha.

Modelo A1:

<KEYBOARD MODEL=A1 INIT\_TYPE=x MASK=yyyyy BEEP=ON SIZE=15 HIDE=OFF ALIGNMENT=ppp RETURN=TEC>, onde x pode ser:

A: vai para teclado maiúsculo  
a: inicia com teclado minúsculo  
9: inicia com teclado numérico

Este modelo contém 3 telas: minúsculos, maiúsculos e numéricos.

MASK = mascara de digitação conforme a regra abaixo:

Na máscara podem ser usados os caracteres:

- 9, x, X, a e A, com as seguintes funções:
- 9 – Só aceita dígitos numéricos.
- a – Só aceita caracteres alfabéticos minúsculos.
- A – Só aceita caracteres alfabéticos maiúsculos.
- x – Aceita caracteres alfabéticos minúsculos e numéricos.

## Especificação de Interface

---

- X – Aceita caracteres alfabéticos maiúsculos e numéricos.

Obs.: O parâmetro MASK sobrescreve o parâmetro SIZE.



### 3.28 SET PORTA SERIAL

Apenas para os modelos que tenham esta saída.

Para enviar informações para a porta serial RS232, devemos passar para o parâmetro TYPE o argumento SERIALINE.

CONTROL especifica a informação que será enviada.

<SET TYPE=SERIALINE CONTROL=X>

Onde X é o valor que será enviado para a porta serial, conforme abaixo:

- 0: -BRK -RTS -DTR
- 1: -BRK -RTS +DTR
- 2: -BRK +RTS -DTR
- 3: -BRK +RTS +DTR
- 4: +BRK -RTS -DTR
- 5: +BRK -RTS +DTR
- 6: +BRK +RTS -DTR
- 7: +BRK +RTS +DTR

### 3.29 CAPTURA ÚNICA DE ENTRADA

Para capturar várias entradas de leitura no mesmo comando usar o comando CAPTURE, informando quais entradas deseja capturar.

Informe o parâmetro NAME, como retorno do comando:

```
<CAPTURE NAME=TESTE RETURN=RETIN>  
</CAPTURE>
```

Para as entradas de captura, não use o parâmetro NAME, pois ele é usado na entrada do comando CAPTURE.

Na variável RETURN (RETIN), é retornado o nome da entrada que foi acionada.

Exemplo:

RETIN = KEYBOARD, para todas as entradas de teclado: FIELD, VALUE, NONINT, PASS, CPF, CNPJ, Mascara genérica, DATA, HOUR.

RETIN = CARD, para entrada de cartão.

RETIN = SERIAL, para entrada serial.

RETIN = TOUCH, para entrada touch.

RETIN = IDCTLS, para entrada ctls.

RETIN = SPECIALKEY, para entrada de teclas especiais.

RETIN = CODEBAR, para entrada USB.

RETIN = MIFARE, para entrada de Mifare.

RETIN = TIMEOUT, quando ocorrer o Send do TIMEOUT.

Exemplo de uso para capturar: entrada digitada, cartão tarja, leitura de código de barras via USB, e Timeout do comando:

```
<CAPTURE NAME=TESTE RETURN=CAPRET>  
  <GET TYPE=FIELD LIN=5 COL=1 SIZE=10>  
  <GET TYPE=CARD TRACKS=2>  
  <GET TYPE=CODEBAR>  
  <GET TYPE=TIMEOUT TIME=10 ACTION=CANCEL>  
</CAPTURE>
```

Onde TIMEOUT pode ser:

<GET TYPE=TIMEOUT TIME=10 ACTION=CANCEL>, indica que se ocorrer um Timeout de 10 segundos, haverá um retorno CANCEL para o início da transação.

Ou

<GET TYPE=TIMEOUT TIME=10 ACTION=SEND VALUE=xxx>, indica que se ocorrer um Timeout de 10 segundos, haverá um retorno da variável TESTE=xxx.

## **Especificação de Interface**

---

OBS.: O uso do TYPE=SPECIALKEY no Capture, junto com TYPE=FIELD, VALUE, PASS, CPF, CNPJ, DATA e HOUR somente tem efeito usando o parâmetro: NOENTER=1.

### 3.30 *INIT*

<INIT KEEP\_COOKIES=xxx>

Define o ponto da transação que o POS retornará quando for acionada a tecla CANCELA (X).  
Onde xxx, pode assumir os seguintes valores : 0 (**não** manter cookies), 1(manter cookies).

### 3.31 *POST*

<POST> Comando para que o POS envie as informações ao Servidor.

A partir da versão 1.9.7 o comando POST pode ser utilizado da seguinte forma :

<POST TIMEOUT=x>, onde :

X é a quantidade de tempo, em segundos, que o POS aguarda a resposta do servidor. Em caso de timeout o POS executa as instruções que vem logo depois do comando.

Obs.: Caso o comando seja utilizado sem o parâmetro TIMEOUT, tudo o que vier depois dele não será executado, como ocorria anteriormente.

A partir da versão 1.18.1, outra opção de post pode ser usada:

<POST RC\_NAME=v IP=i PORT=p RESOURCE=r HOST=h> , onde:

v é a variável que indica o retorno do comando POST caso ele não seja bem sucedido e, nesse caso, os comando seguintes serão executados;

i é o novo IP para ser usado apenas NESTE comando POST;

p é o novo PORT a ser usado apenas NESTE comando POST;

r é o novo RECURSO a ser usado apenas NESTE comando POST;

h é o novo HOST a ser usado apenas NESTE comando POST.

Os parâmetros RC\_NAME, IP, PORT, RESOURCE e HOST são todos opcionais e pode ser usados sozinhos ou em conjunto.

Novo parâmetro no POST a partir da versão 1.34.3:

KCK – keep cancel key

KCK=1 respeita o comando cancel\_key usado anteriormente

KCK=0 anula o efeito do cancel\_key (é o default e como é hoje)

Os seguintes códigos podem ser retornados em um comando POST não bem sucedido:

## Especificação de Interface

---

235 /\* Operation would block \*/  
236 /\* Operation now in progress \*/  
237 /\* Operation already in progress\*/  
238 /\* Socket operation on non-socket \*/  
239 /\* Destination address required \*/  
240 /\* Message too long \*/  
241 /\* Protocol wrong type for socket \*/  
242 /\* Protocol not available \*/  
243 /\* Protocol not supported \*/  
244 /\* Socket type not supported \*/  
245 /\* Operation not supported \*/  
246 /\* Protocol family not supported \*/  
247 /\* Address family not supported by protocol family \*/  
248 /\* Address already in use \*/  
249 /\* Can't assign requested address \*/  
250 /\* Network is down \*/  
251 /\* Network is unreachable \*/  
252 /\* Network dropped connection on reset \*/  
253 /\* Software caused connection abort \*/  
254 /\* Connection reset by peer \*/  
255 /\* No buffer space available \*/  
256 /\* Socket is already connected \*/  
257 /\* Socket is not connected \*/  
258 /\* Can't send after socket shutdown \*/  
259 /\* Too many references: can't splice \*/  
260 /\* Operation timed out \*/  
261 /\* Connection refused \*/  
264 /\* Host is down \*/  
265 /\* No route to host \*/

### 3.32 FILE

Para receber um arquivo do servidor, usar o comando abaixo:

```
<FILE NAME=xxxxx ADDR=yyyyy ERR=zzzzz WRT=www>
```

Onde :

xxxxx é o nome do arquivo que será gravado na RAM do POS.

yyyyy é o nome do diretório e nome do arquivo origem no Servidor.

Zzzzz é o nome da variável que receberá o resultado da operação.

Wwww é um parâmetro enviado que decide que se já houver o arquivo sobrescreve ou não o mesmo, enviar: "SIM" se deseja sobrescrever arquivo e "NÃO" se não deseja.

A variável ERR é retornada ao Servidor com o status do erro, se 0 foi sucesso, se 1 deu erro e se 2 não sobrescreveu porque já havia arquivo.

Exemplo:

```
<FILE NAME=logoprint.bmp ADDR=/logoprint.bmp ERR=ERROARQ WRT=SIM>
```

O retorno virá na variável ERROARQ com os valores descritos.

#### Observação 1 :

Com este comando também é possível fazer a atualização remota da versão do NAVS. Vide capítulo 3.9.1 TELECARGA VIA COMANDO FILE.

#### Observação 2 :

Para o comando funcionar em servidores não HTML puros, é preciso programar o comando response abrindo o arquivo que vem no request e o enviando pelo comando response.

**A seguir um exemplo de Controller JAVA para o comando FILE :**

## Especificação de Interface

---

```
@WebServlet("/DOWNLOAD")
```

```
public class ControllerServlet extends HttpServlet {
```

```
    protected void service(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
```

```
        String file = request.getQueryString();
```

```
        ServletContext context = getServletContext();
```

```
        String filename = context.getRealPath(file);
```

```
        File f = new File(filename);
```

```
        int length = 0;
```

```
        ServletOutputStream op = response.getOutputStream();
```

```
        String mimetype = context.getMimeType( file );
```

```
        // Set the response and go! :
```

```
        response.setContentType( (mimetype != null) ? mimetype : "application/octet-stream" );
```

```
        response.setContentLength( (int)f.length() );
```

```
        // Stream to the requester :
```

```
        byte[] bbuf = new byte[2000];
```

```
        DataInputStream in = new DataInputStream(new FileInputStream(f));
```

```
        while ((in != null) && ((length = in.read(bbuf)) != -1)) {
```

```
            op.write(bbuf,0,length);
```

```
        }
```

```
        in.close();
```

```
        op.flush();
```

```
        op.close();
```

```
    }
```

```
}
```

### Nova Sintaxe do Comando FILE para este caso:

<FILE NAME=imagem.bmp ADDR=/aaa/DOWNLOAD?imagens/imagem.bmp ERR=ERRO WRT=NAO>

Onde :

“/aaa/DOWNLOAD” é o recurso no seu servidor que responderá pelo download.

“?” informa para o controlador qual é o parâmetro passado.

E “/imagem.bmp” é o path e o arquivo dentro do contexto do seu servidor.



### **3.33 REMOVE ARQUIVO NO POS**

Para apagar um arquivo no POS, usar o comando abaixo:

```
<DELFILE NAME=xxxxx ERR=zzzzz>
```

Onde,

xxxxx é o nome do arquivo que será removido no POS.

zzzzz é o nome da variável que receberá o resultado da operação.

A variável ERR é retornada ao Servidor com o status da operação, se 0 foi sucesso, se 1 foi erro.

Exemplo:

```
<DELFILE NAME=logoprint.bmp ERR=ERROARQ>
```

### 3.34 DUAL CHIP

Para usar o recurso Dual Chip em POS que tenham esta característica, usar o comando abaixo:

SIM1, SIM2, OTHER DEFAULT

<GPRS\_SWITCH\_SIM xxxxx>

Onde,

xxxxx pode ser:

SIM1 - chaveia para o SIM 1

SIM2 - chaveia para o SIM 2

OTHER - chaveia para o outro, se está com SIM 1 vai para o SIM 2 e vice-versa

DEFAULT - chaveia para o SIM 1

Exemplo:

<GPRS\_SWITCH\_SIM SIM1>

### 3.35 COPIA DE/PARA DISPOSITIVOS EXTERNOS - ARQUIVOS

Efetua copia de arquivos de ou para dispositivos externos, como Pendrive e Memória SD. Para apagar um arquivo no POS, usar o comando abaixo:

<DEVICE TYPE=xxxxx NAME=yyyyy ERR=zzzzz>

Onde,

xxxxx é o tipo da função, que pode ser:

COPYTO: copia arquivo da memória do POS para o dispositivo externo.

COPYFROM: copia arquivo do dispositivo externo para a memória do POS.

DELETE: apaga arquivo do dispositivo externo.

Yyyyy é o nome do arquivo e dispositivo externo.

“M.”, dispositivo Pendrive

“L.”, dispositivo Memória SD

zzzzz é o nome da variável que receberá o resultado da operação.

A variável ERR é retornada ao Servidor com o status da operação, se 0 foi sucesso, se 1 foi erro.

Exemplo:

<DEVICE TYPE=COPYTO NAME=M:init.xml ERR=ERROARQ>:

Copia o arquivo “init.xml” do POS para o Pendrive.

<DEVICE TYPE=COPYFROM NAME=M:arquivo.lua ERR=ERROARQ>:

Copia o arquivo “arquivo.lua” do Pendrive para o POS.

<DEVICE TYPE=DELETE NAME=M:lixo.txt ERR=SDELF>:

Apaga o arquivo “lixo.txt” do Pendrive.

### **3.36 *RESET***

<RESET> Comando para reiniciar o POS.

### 3.37 CONFIG\_NAVS

Comando que altera ou recebe as configurações NAVS remotamente.

Em caso de erro a variável que for indicada no parâmetro RETURN receberá o código de erro.

**Abaixo um exemplo de uso do comando para enviar as configurações:**

```
<CONFIG_NAVS RETURN=STS_ALTERA_SERVER>
  PASSWORD_CONFIG=senha
  NEW_PASSWORD_CONFIG=novasenha
  CONECTION_TYPE=E;
  WI_FI_SSID=skytefwifi;
  WI_FI_PASSWORD=skytef;
  LOCAL_IP=192.168.1.58;
  LOCAL_MASK=255.255.0.0;
  LOCAL_GATEWAY=192.168.0.1;
  LOCAL_DNS_1=192.168.0.103;
  LOCAL_DNS_2=192.168.0.103;
  LOCAL_PING=8;
  GPRS_CONFIG=1;
  GPRS_APN=ZAP.VIVO.COM.BR;
  GPRS_USER=VIVO;
  GPRS_PASSWORD=VIVO;
  SERVER_IP=200.160.80.90;
  SERVER_PORT=6789;
  SERVER_RESOURCE=/TESTE.PHP;
  SERVER_HOST=200.160.80.90;
  SERVER_HTTPS_ACTIVE=0;
  SERVER_HTTPS_METHOD=1;
  POSITION_STATUS_LINE=B;
  SHOW_HOUR_AT_STATUS_LINE=1;
  SCROLL_UP=62;
  SCROLL_DOWN=63;
  PRINTER_CONTRAST=1;
  KEEP_ALIVE_ATIVAR=S;
  KEEP_ALIVE_TEMPO_DE_INTERVALO=20;
  KEEP_ALIVE_IP_DESTINO=200.160.80.90;
  KEEP_ALIVE_PORT=6789;
  BAUDRATE_SERIAL=28800;
  PARIDADE_SERIAL=PAR;
  DATA_BITS_SERIAL=7;
  STOP_BITS_SERIAL=1;
  TIMEOUT_SERIAL=5;
  SENHA_SUP=1234;
  *BAT1PERCENT=1;
  *OFF=1800;
  *DSP_BL_DLY=4;
  *POW=15000;
```

## Especificação de Interface

---

```
NAVS_ECONOMIA_BATERIA=S;  
IP_SL=SKYTEFNAVS.NOIP.ME;  
PORT_SL=8090;  
RESOURCE_SL=/LICENCAONLINE/POS;  
HOST_SL=SKYTEFNAVS.NOIP.ME;  
PGM_SSL_CLI=0;
```

//novos parâmetros a partir da versão 1.18.7:

```
GPRS_SIM2_CONFIG=  
GPRS_SIM2_APN=  
GPRS_SIM2_USR=  
GPRS_SIM2_PASSWORD=  
WIFI_LIST=  
TRANS_PEND=  
NAVS_COMM_GPRS_HS_NETWORK=  
PGM_RECIBO=  
NAVS_MSGS_LOL=
```

//novos parâmetros a partir da versão 1.18.7:

```
GPRS_SIMS=  
CONNECTION_TYPE_2=E, G ou W  
DISABLE_GPRS=
```

//novos parâmetros a partir da versão 1.32.0:

```
FORCA_DCNX=
```

//novos parâmetros a partir da versão 1.36.0:

```
NGPS=  
NGPSTIME=
```

</CONFIG\_NAVS>

Obs.: Não é mais obrigatória a inserção de todos os parâmetros, basta inserir o parâmetro que se deseja alterar e inserir o parâmetro, seguindo o padrão acima demonstrado.

Obs.2: Para reiniciar o POS, após indicar os novos valores é necessário utilizar o comando <RESET>.

Obs.3: Somente quando “SERVER\_IP” e “SERVER\_PORT” o POS é reiniciado sem a necessidade de uso do comando <RESET>.

Obs. 4: Para alterar os parâmetros é necessário que a senha (PASSWORD\_CONFIG) esteja correta.

Valores que cada parâmetro pode receber:

## **Especificação de Interface**

---

PASSWORD\_CONFIG, senha atual do dispositivo que se deseja alterar os parâmetros, deverá ser informado sempre.

NEW\_PASSWORD\_CONFIG, nova senha, se deseja alterar a atual.

CONNECTION\_TYPE, define que tipo de conexão será utilizada pelo POS E(ETHERNET), G(GPRS), W(WIFI). É importante lembrar que a configuração só será aplicada se o POS suportar o tipo de conexão aferido.

WI\_FI\_SSID, recebe o SSID da rede em que se deseja inserir o POS.

WI\_FI\_PASSWORD, recebe a senha da rede Wi-Fi.

LOCAL\_IP, recebe o IP local, se utilizar DHCP preencha com 0.0.0.0.

LOCAL\_MASK, recebe a mascara da rede.

LOCAL\_GATEWAY, recebe o GATEWAY da rede.

LOCAL\_DNS\_1, recebe o valor de DNS, primeira opção.

LOCAL\_DNS\_2, recebe o valor de DNS, segunda opção.

LOCAL\_PING, recebe o valor de ping.

GPRS\_CONFIG, define qual a configuração de chip (APN, USUÁRIO, SENHA), serão utilizados. Se receber 1, esses valores terão de ser enviados ao servidor, pois esse caso representa uma exceção aos padrões definidos. Se igual a 2, o padrão adotado será da operadora VIVO, se igual a 3, o padrão adotado será da operadora CLARO, se igual a 4, o padrão adotado será da operadora OI, e se igual a 5, o padrão adotado será da operadora TIM.

GPRS\_APN, recebe a APN da operadora, só deverá ser preenchido se GPRS\_CONFIG receber valor igual a 1.

GPRS\_USER, recebe o nome de usuário da operadora, só deverá ser preenchido se GPRS\_CONFIG receber valor igual a 1.

GPRS\_PASSWORD, recebe a senha de usuário da operadora, só deverá ser preenchido se GPRS\_CONFIG receber valor igual a 1.

SERVER\_IP, recebe o IP do servidor onde estará o recurso NAVS a ser acessado.

SERVER\_PORT, recebe o id da porta do servidor onde estará o recurso NAVS a ser acessado.

SERVER\_RESOURCE, recebe o nome do recurso a ser acessado, lembrando que "/" deverá estar antes do nome (Ex.: "/TESTE.PHP").

SERVER\_HOST, recebe o HOST do servidor a ser acessado.

## Especificação de Interface

---

SERVER\_HTTPS\_ACTIVE, define se a conexão será https enviando valor igual a 1, ou http enviando valor igual a 0.

SERVER\_HTTPS\_METHOD, recebe definição do tipo de conexão https. 1=SSLv3, 2=TLSv1. Poderá receber um valor nulo somente se SERVER\_HTTPS\_ACTIVE = 0.

POSITION\_STATUS\_LINE, define a posição da linha de status, e recebe os valores "T"(topo), "B"(rodapé), "N"(ausente).

SHOW\_HOUR\_AT\_STATUS\_LINE, define se a hora irá ou não aparecer na linha de status. (0=ocultar, 1=mostrar).

SCROLL\_UP, receberá os valores que são definidos para essa tecla. (Verificar referencia ao parâmetro SCROLL nesta mesma documentação). Não poderá ter o mesmo valor que SCROLL\_DOWN.

SCROLL\_DOWN, receberá os valores que são definidos para essa tecla. (Verificar referencia ao parâmetro SCROLL nesta mesma documentação). Não poderá ter o mesmo valor que SCROLL\_UP.

PRINTER\_CONTRAST, define o nível de contraste para a impressão, assume valores de 1 a 8.

KEEP\_ALIVE\_ATIVAR, define se o keep alive estará ou não ativo, assume os valores "S"(sim) e "N"(não).

KEEP\_ALIVE\_TEMPO\_DE\_INTERVALO, define a frequência de envio em segundos do número de série ao servidor e porta programados.

KEEP\_ALIVE\_IP\_DESTINO, recebe o IP de destino onde o número de série do POS será enviado.

KEEP\_ALIVE\_PORT, recebe a porta do IP de destino onde o número de série do POS será enviado.

BAUDRATE\_SERIAL, recebe o valor de Baud Rate da serial, onde os valores permitidos são : "300", "600", "1200", "2400", "4800", "9600", "19200", "38400", "57600", "115200", "12000", "14400", "28800", "33600". Qualquer valor fora desse range não será aceito, fazendo com que o POS retorne o erro.

PARIDADE\_SERIAL, recebe a definição de paridade da serial. Os valores aceitos são : "PAR", "IMPAR" e "NENHUMA". Qualquer valor fora desse range não será aceito, fazendo com que o POS retorne o erro.

DATA\_BITS\_SERIAL, recebe a definição de data bits da serial. Os valores aceitos são : "7" e "8". Qualquer valor fora desse range não será aceito, fazendo com que o POS retorne o erro.

STOP\_BITS\_SERIAL, recebe a definição de stop bits da serial. Os valores aceitos são : "1", "2". Qualquer valor fora desse range não será aceito, fazendo com que o POS retorne o erro.

TIMEOUT\_SERIAL, define o tempo de timeout da serial.

SENHA\_SUP, define a senha do supervisor para o comando pagamento.



## Especificação de Interface

---

\*BAT1PERCENT, define 1%(1) ou 10% (0) de incremento/decremento no status da carga da bateria.

\*OFF, define o tempo em segundos para o POS desligar após entrar no modo sleep (dormir).

\*DSP\_BL\_DLY, define o tempo em segundos para diminuir a luminosidade do display do POS antes de entrar em sleep.

\*POW, define o tempo em milissegundos para o POS entrar no modo sleep quando estiver sem uso.

NAVS\_ECONOMIA\_BATERIA, define se o POS entra no modo de economia de bateria (onde os 3 parâmetros anteriores tem validade), pode ser S ou N.

IP\_SL, define o IP de acesso ao servidor LOL.

PORT\_SL, define a Porta de acesso ao servidor LOL.

RESOURCE\_SL, define o Recurso de acesso ao serviço LOL.

HOST\_SL, define o HOST do servidor de acesso ao serviço LOL, se necessário.

PGM\_SSL\_CLI, define o uso do SSL para o Pagamento: 1 = Ativado, 0 = Desativado.

GPRS\_SIM2\_CONFIG, idem GPRS\_CONFIG para o SIM 2 (modelos que tem 2 chips).

GPRS\_SIM2\_APN, idem GPRS\_APN para o SIM 2.

GPRS\_SIM2\_USR, idem GPRS\_USR para o SIM 2.

GPRS\_SIM2\_PASSWORD, idem GPRS\_PASSWORD para o SIM 2

WIFI\_LIST, modo de operação com a lista de redes WiFi:

0-Desativar: vale o que está configurado e não exibe a lista.

1-Exibir sempre: vai exibir a lista das redes sempre que o POS foi iniciado.

2-Exibir somente em erros: vai exibir a lista somente em casos de erros de conexão.

3-Exibir apenas uma vez: vai exibir a lista somente 1 vez.

4-Exibir apenas uma vez (\*): a lista é exibida apenas 1 vez, ou sempre que ocorrer erro de conexão

TRANS\_PEND, modo de operação para transações pendentes: 1 = Desfaz, 2 = Confirma

NAVS\_COMM\_GPRS\_HS\_NETWORK, modo de operação do GPRS: 0 = default; 1 = automático; 2 = somente 2G; 3 = somente 3G.

PGM\_RECIBO, impressão dos recibos: 0 = Normal; 1 = Ampliado; 3 = Não Imprime.

NAVS\_MSGS\_LOL, mostra mensagens do LOL: S = Sim; N = Não.

GPRS\_SIMS: 1 = UNO - SIM1; 2 = UNO - SIM2; 3 = DUAL MANUAL; 4 = DUAL AUTOMATICO.

CONNECTION\_TYPE\_2, define que tipo de conexão será utilizada pelo POS E(ETHERNET), G(GPRS), W(WIFI). É importante lembrar que a configuração só será aplicada se o POS suportar o tipo de conexão aferido.

## Especificação de Interface

DISABLE\_GPRS, desativa o GPRS quando simcard ausente: S ou N.

FORCA\_DCNX, força a desconexão com o Sitef no final da transação: 1 ou 0, padrão é 1.

NGPS, ativa ou desativa o GPS, 0 ou 1, padrão é 0.

NGPSTIME, tempo em minutos para ler o GPS, de 1 a 999.

RETURN deve receber a variável que terá um retorno do status da operação.

A partir da versão 1.8.8 os valores de retorno foram mudados.

A resposta será "OK" se tudo estiver correto, ou uma string com o(s) nome(s) do(s) parâmetro(s) que apresenta(m) erro.

### Abaixo como usar o comando para receber as configurações atuais do NAVS:

```
<CONFIG_NAVS GET RETURN=INFO>
  PASSWORD_CONFIG=senha;
</CONFIG_NAVS>
```

A senha deve estar correta, caso contrário às informações não serão retornadas.

GET é o parâmetro que indica que as informações deverão ser retornadas e não modificadas.

O retorno será uma string, na variável definida em RETURN, como por exemplo, essa abaixo:

"P1=E,P2=skytefwifi,P3=skytef,P4=192.168.1.58,P5=255.255.0.0,P6=192.168.0.1,P7=192.168.0.103,P8=192.168.0.103,P9=15,P10=1,P11=ZAP.VIVO.COM.BR,P12=VIVO,P13=VIVO,P14=200.160.80.90,P15=6789,P16=/TESTE.PHP,P17=200.160.80.90,P18=0,P19=1,P20=B,P21=1,P22=62,P23=63,P24=1,P25=S,P26=20,P27=200.160.80.90,P28=6789,P29=28800,P30=PAR,P31=7,P32=1,P33=5,P34=v1.9.1,P35=QT6Z0303,P36=1.11.4.10,P37=1.08v4.48<sup>a</sup>140718,P38=6.2.0,P39=4.0.111.14.r1,P40=1234,P41=1,P42=1800,P43=15000,P44=4,P45=S,P46=SKYTEFNAVS.NOIP.ME,P47=8090,P48=/LICENCAONLINE/POS,P49=,P50=0,....."

A seguir a tabela de equivalência dos parâmetros representados por Px, onde X é a numeração :

ID Parâmetro	Parâmetro Correspondente
P1	CONNECTION_TYPE
P2	WI_FI_SSID
P3	WI_FI_PASSWORD
P4	LOCAL_IP
P5	LOCAL_MASK
P6	LOCAL_GATEWAY
P7	LOCAL_DNS_1
P8	LOCAL_DNS_2
P9	LOCAL_PING
P10	GPRS_CONFIG
P11	GPRS_APN
P12	GPRS_USER
P13	GPRS_PASSWORD
P14	SERVER_IP
P15	SERVER_PORT
P16	SERVER_RESOURCE

## Especificação de Interface

P17	SERVER_HOST
P18	SERVER_HTTPS_ACTIVE
P19	SERVER_HTTPS_METHOD
P20	POSITION_STATUS_LINE
P21	SHOW_HOUR_AT_STATUS_LINE
P22	SCROLL_UP
P23	SCROLL_DOWN
P24	PRINTER_CONTRAST
P25	KEEP_ALIVE_ATIVAR
P26	TIME_KEEP_ALIVE
P27	KEEP_ALIVE_IP_DESTINO
P28	KEEP_ALIVE_PORT
P29	BAUDRATE_SERIAL
P30	PARIDADE_SERIAL
P31	DATA_BITS_SERIAL
P32	STOP_BITS_SERIAL
P33	TIMEOUT_SERIAL
P34	VERSAO_NAVS
P35	VERSAO_SO
P36	VERSAO_EOS
P37	VERSAO_BC

ID Parâmetro	Parâmetro Correspondente
P38	VERSAO_EMV
P39	VERSAO_CLISITEF
P40	SENHA_SUP
P41	*BAT1PERCENT
P42	*OFF
P43	*POW
P44	*DSP_BL_DLY
P45	NAVS_ECONOMIA_BATERIA
P46	IP_SL
P47	PORT_SL
P48	RESOURCE_SL
P49	HOST_SL
P50	PGM_SSL_CLI
P51	GPRS_SIM2_CONFIG
P52	GPRS_SIM2_APN
P53	GPRS_SIM2_USR
P54	GPRS_SIM2_PASSWORD
P55	WIFI_LIST
P56	TRANS_PEND
P57	NAVS_COMM_GPRS_HS_NETWORK
P58	PGM_RECIBO
P59	NAVS_MSGS_LOL
P60	GPRS_SIMS
P61	CONNECTION_TYPE_2
P62	DISABLE_GPRS
P63	FORCA_DCNX
P64	NGPS
P65	NGPSTIME

### 3.38 IMPRESSÃO DE BOLETO

<BOLETO BOLRET=xxxxx> inicia texto para o boleto,  
</BOLETO> finaliza texto para o boleto.

Exemplo de uso com alguns campos apenas:

<BOLETO BOLRET=STSBOL> inicia texto para o boleto,  
NOME\_BANCO=Bradesco;  
BANCO\_NUM=001;  
</BOLETO > finaliza texto para o boleto.

IMPORTANTE: os campos devem sempre finalizar com ';'.

A variável BOLRET é retornada ao Servidor com o status da impressão do Boleto.  
As seguintes strings podem retornar:

OK  
CODIGO INVALIDO  
ERRO CODIGO  
SEM PAPEL  
ERRO IMPRESSAO  
ERRO DESCONHECIDO

Abaixo todos os campos do boleto, com conteúdo de exemplo:

NOME\_BANCO=Bradesco;  
BANCO\_NUM=001;  
BANCO\_DIG=9;  
NOSSO\_NUM=0000001254588;  
VALOR\_EM\_CENTAVOS=12345678;  
VALOR\_DOCUMENTO\_EM\_CENTAVOS=8999999123;  
DESC\_EM\_CENTAVOS=1;  
DED\_EM\_CENTAVOS=22;  
MULTA\_EM\_CENTAVOS=333;  
ACRESC\_EM\_CENTAVOS=4444;  
VALOR\_COBRADO\_EM\_CENTAVOS=11122233344;  
VENCIMENTO=16/07/2012;  
MOEDA=9;  
LOCAL=qualquer banco até o vencimento, após o vencimento apenas em agências XXX;  
CEDENTE=Software Express (teste);  
DATA\_DOC=14/07/2012;  
NUMERO\_DOC=678;  
ESPECIE\_DOC=DM CI;  
ACEITE=N;  
DATA\_PROC=15/07/2012;  
USO\_BANCO=0.000.000/27272;

## Especificação de Interface

---

CARTEIRA=018;  
QUANTIDADE=1;  
INSTR1=Não receber após 30 dias da data de vencimento.;  
INSTR2=Instruções linha 2;  
INSTR3=Instruções linha 3;  
INSTR4=Instruções linha 4;  
INSTR5=Instruções linha 5;  
SACADO1=Teste;  
SACADO2=Av. Paulista, 2202 Sobreloja H;  
SACADOR=abcde;  
AGENCIA\_COD\_CEDENTE=3376-4/12678-0;  
CONTRATO=00.00.00;  
CPF=20.774.232/0001-34;  
CAMPO\_LIVRE=6666666777777777777777775555;

Obs.: Com nova versão da LIB Boletto (atualizada dia 06/06/2014), é possível “zerar” o fator de vencimento na linha digitada se a data de vencimento fornecida for um texto qualquer. Exemplo de valores para os parâmetros:

VALOR\_EM\_CENTAVOS=0;  
VALOR\_DOCUMENTO\_EM\_CENTAVOS=0;  
VALOR\_COBRADO\_EM\_CENTAVOS=0;  
VENCIMENTO=contra 101performance101;

### 3.39 ENVIO DE ARQUIVO

<SENDFILE NAME=XXXXX > comando para enviar arquivo para o Servidor.

O parâmetro NAME informa o nome do arquivo a ser enviado.

Qualquer tipo de arquivo pode ser enviado, texto ou não texto.

Abaixo, um exemplo em PHP, de como o Servidor deve capturar o arquivo recebido:

```
<?php
var_dump ($_FILES);
copy ($_FILES["file"]["tmp_name"], "C:\\wamp\\www\\".$_FILES["file"]["name"]);

$tela = $_GET["TELA"];
switch ($tela){
case 1:
    echo "<GET TYPE=HIDDEN NAME=TELA VALUE=2>";
    echo "<CONSOLE>";
    echo "<BR><BR>ENTER PARA ENVIAR";
    echo "</CONSOLE>";
    echo "<GET TYPE=ANYKEY>";
    echo "<SENDFILE NAME=claro.bmp>";

break;
}
?>
```

Importante: Este comando substitui o POST, ou seja, ele é o último comando que deve ser inserido na lista da sessão.

Nenhum outro comando após ele será executado na sessão.

### 3.40 UNZIP DE ARQUIVO NO POS

Para unzipar um arquivo no POS, usar o comando abaixo:

```
<UNZIP NAME=xxxxx ERR=zzzzz>
```

Onde,

xxxxx é o nome do arquivo que será unzipado no POS.

zzzzz é o nome da variável que receberá o resultado da operação.

A variável ERR é retornada ao Servidor com o status da operação, se 0 foi sucesso, se 1 foi erro.

Exemplo:

```
<UNZIP NAME=TESTE.ZIP ERR=ERROARQ>
```

Após unzipar corretamente o arquivo, este é deletado automaticamente do POS.

Se ocorrer erro no UNZIP, o arquivo é mantido.

### **3.41 COMANDOS AND, OR, XOR, URL\_encode e URL\_decode para LUA**

\_bw\_and\_8 – executa um AND de 8 bits

\_bw\_or\_8 – executa um OR de 8 bits

\_bw\_xor\_8 – executa um XOR de 8 bits

\_bw\_and\_16 – executa um AND de 16 bits

\_bw\_or\_16 – executa um OR de 16 bits

\_bw\_xor\_16 – executa um XOR de 16 bits

Todos recebem 2 variáveis numéricas como parâmetros e retornam o resultado em uma variável numérica:

Forma de chamada:

c=\_bw\_and\_8(a,b)

c=\_bw\_or\_8(a,b)

c=\_bw\_xor\_8(a,b)

c=\_bw\_and\_16(a,b)

c=\_bw\_or\_16(a,b)

c=\_bw\_xor\_16(a,b)

x=\_URL\_encode(area)

x=\_URL\_decode(area)



### 3.42 *HTMLDEBUG*

Para verificar tudo o que vem do Servidor para o POS, faz a impressão do log de toda a instrução enviada.

<HTMLDEBUG ON> ativa a impressão do log.

<HTMLDEBUG OFF> desativa a impressão do log.

### 3.43 *RELICENCIA*

Para relenciar o POS, use o comando:.

<RELICENSE NOW> remove a licença do NAVS do POS, reinicia o POS para solicitar a licença.

<RELICENSE> remove a licença do NAVS do POS. No próximo reinício, a licença será solicitada.

### **3.44 GRAVAÇÃO EM ARQUIVO DAS CAPTURAS – POSTOFF (uso para offline)**

<POSTOFF NAME=XXXXX> comando para que o POS grave no arquivo indicado no parâmetro NAME as informações coletadas.

Este comando grava um registro no arquivo informado, com as informações dos dados coletados pelos comandos anteriores ao POSTOFF.

Após a gravação do registro, o NAVS retorna para a posição do INIT.

O formato do arquivo é o mesmo do dado enviado ao Servidor pelo comando POST.

Exemplo: nome1=campo1&nome2=campo2&

Os registros são separados pelo caracter '%’.

### 3.45 IMPRESSÃO OFFLINE – PRINTEROFF

Este comando executa a impressão dos dados como no comando PRINTER, com a possibilidade de imprimir os dados coletados offline (até o comando POSTOFF).

<PRINTEROFF> inicia texto para a impressora,  
</PRINTEROFF> finaliza texto para a impressora,

A mudança de linha é realizada pelo comando <BR>.

Para imprimir dados coletados offline, use o nome da variável entre o caracter '%’.

Exemplo: “S/N: %SERIE%”

### 3.46 **VALIDAÇÃO OFFLINE – VALIDATEOFF**

< VALIDATEOFF NAME=XXXXX> comando para validar as informações capturadas Offline.

O parâmetro NAME informa o nome do arquivo com os dados a serem validados com as informações coletadas até o POSTOFF.

Cada dado deve ser separado pelo caracter: '% '.

Exemplo do conteúdo deste arquivo:

%528922219%VX520%5327517780006%

Todos os dados coletados, deverão estar dentro deste arquivo.

No exemplo acima,

Caso a validação esteja correta, o NAVS prossegue para o próximo comando, que poderá ser a impressão Offline.

Caso a validação seja negada, ou seja, algum dado coletado não se encontra no arquivo informado, o NAVS apaga as informações coletadas Offline e retorna para o comando INIT.

## Especificação de Interface

---

### EXEMPLO OFFLINE:

```
<INIT>
<GET TYPE=SERIALNO NAME=SERIE>
<GET TYPE=MODEL NAME=MODPOS>

<CONSOLE><BR><BR>DIGITE O RG</CONSOLE>
<GET TYPE=FIELD NAME=NUMRG SIZE=9 COL=0 LIN=5>

<VALIDATEOFF NAME=ARQVAL>

<PRINTEROFF>
TRANSAÇÃO APROVADA OFFLINE<BR><BR>
S/N: %SERIE%<BR>
MODELO: %MODPOS%<BR>
RG: %NUMRG%<BR>
</PRINTEROFF>

<POSTOFF>
```

Na impressora sairá a informação abaixo, caso a validação dos dados esteja correta (VALIDATEOFF):

### TRANSAÇÃO APROVADA OFFLINE

S/N: 521000999  
MODELO: VX520  
RG: 12123433-2

A seguinte informação será gravada no arquivo:

**SERIE=521000999&MODPOS=VX520&NUMRG=12123433-2&**

Supondo que no arquivo ARQVAL informado no comando VALIDATEOFF, contenha os dados abaixo:  
%528123456%VX520%105554446%999555476%123456789%

O NAVS aceitará a captura, aprovando a transação e imprimindo os dados, caso o numero de série do terminal seja o 528123456, e o modelo do POS seja o VX520 e o RG coletado seja: 105554446, ou 999555476 ou 123456789.

Para remover arquivos Offline do POS, use o comando DELFILE (capítulo 3.16):

```
<DELFILE NAME=xxxxxx ERR=zzzzz>
```

### 3.47 TELECARGA

#### 1. VIA COMANDO <FILE> OU <FTP>

Para efetuar a Telecarga remota de uma versão mais atualizada via comando FILE, coloque a nova versão no Servidor com o nome NAVS.ZIP, e, use a sequência de comandos abaixo para iniciar a transferência do arquivo e efetivar a atualização no POS:

```
<FILE NAME=navs.zip ADDR=/navs.zip ERR=ERROARQ>  
<RESET>
```

O comando FILE irá processar a transferência do arquivo e o comando RESET irá reiniciar o POS para que a versão possa ser atualizada no POS.

Abaixo o comando para uso com FTP:

```
<FTP TYPE=GET HOST=b PORT=c USERID=d PASSW=e LOCALFILE=l:navs.zip REMOTEFILE=navs.zip  
RETURN=H>  
<RESET>
```

#### 2. VIA SERVIDOR VERICENTRE (EM DESUSO)

Para efetuar a Telecarga remota de uma versão mais atualizada usar o comando abaixo:

```
<TELECARGA ZA=NAVS ZT=555 ADDRESS=200.160.80.90 PORT=8013 TPCARGA=W>
```

Onde,

NAVS (za) é um exemplo de nome da aplicação no Servidor de Telecargas Vericentre.

555 (zt) é um exemplo da identificação do terminal no Servidor Vericentre.

200.160.80.90 (address) é o IP do Servidor Vericentre.

8013 (port) é a porta do Servidor Vericentre.

W (tpcarga) é o tipo de telecarga, que pode ser:

F – carga total com Reset do POS (apaga a carga atual)

P – carga parcial com Reset do POS.

p – carga parcial sem Reset do POS.

## 4.0 OFFLINE com Lua / SQLite

Desenvolvimento de transações no NAVS em modo Offline, agora de uma forma mais ampla, pode ser efetuado através da linguagem LUA com o banco de dados SQLite intergados.

Para acionar o script lua é preciso utilizar o comando data, exemplificado abaixo:

```
<data destination=parser>  
    dofile("\NomeDoArquivoLua.lua")  
    SuaFuncaoNoArquivoLua()  
</data>
```

Dentro de um script lua é possível chamar outro módulo lua do mesmo modo descrito acima através do comando “data”.

Para obter o valor das variáveis carregadas antes do uso do comando **data** no lua, basta carregar o conteúdo da variável GET utilizando o loadstring, conforme demonstrado abaixo :

```
local get = loadstring(GET)()
```

E para obter o valor da variável basta seguir o exemplo abaixo :

```
local Var = get[“VARIÁVEL”]
```

Para retornar ao servidor é preciso utilizar o comando POST.

Para fazer debug do LUA utilizando a porta serial do POS, utilize o seguinte comando no script lua:

`_debug_on()` - para ativar o debug sob o LUA  
`_debug_off()` – para desativar o debug sob o LUA  
`_logserial(var)` ou `_logserial(“xxx”)`  
`_funName()` – para retornar o nome da função LUA sendo executada  
`_callerName()` – para retornar o nome da função LUA chamadora da função que executou esse comando.

`_logserial(“teste”)`; ou `_logserial(StringExemplo)`, onde “StringExemplo” é uma string que recebeu algum valor em algum ponto do script LUA desenvolvido; e “teste” é uma string literal.

Tudo que estiver dentro de “`_logserial(“xxxx”)`”, ou seja, tudo o que for “xxxx””, será enviado para a serial do POS.

Obs.: Ao criar arquivos com extensão “db” não ultrapassar 8 bytes no tamanho do nome do arquivo, sem contar com a identificação da extensão (.db).

Obs2.: A sintaxe do comando foi alterada de “`debug(xxx)`” para “`_logserial(xxx)`”.



## Especificação de Interface

---

Bibliotecas integradas no NAVS: navs.o e comunica.o

Navs.o com as funções:

- navs.xml(file, ...)

Executa os comandos navs de um arquivo xml.

file : nome do arquivo a ser executado.

... : Parametros sequencias que podem ser acessados no XML atraves da marcação ("**LUA\_PAR**").

- navs.call(file, proced)

chama uma função de um arquivo lua (call\_module).

file : nome do arquivo lua que contém a função.

proced : Função a ser executada

- navs.post()

execulta um comando <post> do navs.

Comunica.o com as funções:

- prepare()

Prepara o socket para operações de comunicação

\* retorna erro caso obtenha timeout

- connect(ip, port)

Conecta um socket via TCP à um ip e porta.

ip : ip a ser cnectado

port : porta a ser conectada

\* retorna erro caso haja erro de conexão

- disconnect()

Libera o socket de uma conexão

- send(tosend)

envia uma mensagem para um socket pré-estabelecido

## Especificação de Interface

---

tosend : mensagem a ser enviada

\* retorna erro caso haja algum

- receive()

recebe uma mensagem de um socket pré-estabelecido

\* retorna a mensagem recebida ou um erro caso haja

- getHttp(recurso, ...)

realiza um GET via HTTP usando um socket pré-estabelecido

recurso : Nome do recurso que receberá o GET

... : Parametros que podem ser passados pelo GET

formato: "NOME=VALOR"

\* retorna erro se houver

- receiveHttp()

recebe uma mensagem via HTTP

\* retorna a mensagem recebida

## 5.0 Compilador Lua

Essa ferramenta serve para não ter a necessidade de colocar todos os seus scripts .lua no POS e para mantê-los mais seguros.

Ela é um pacote zip que deverá ser extraído e que contém os seguintes arquivos:

- "ChunkSpy.lua";
- "lua.bat";
- "lua.exe"
- "luac.exe";

Para gerar arquivo compilado lua, é necessário alterar um único arquivo desse pacote : o "lua.bat".

Exemplo para "lua.bat":

```

1 C:
1 cd "C:\Users\Skytef\Desktop\Minhas Ferramentas\Compilador Lua"
1 luac -o temp.o -l ^
1 script1.lua ^
1 script2.lua ^
1 script3.lua
7
8
9 pause
10 lua ChunkSpy.lua temp.o --rewrite "arm standard" -o ArqFinal.o
  
```

Onde :

- na linha 2 deverá ter o endereço da pasta onde esses arquivos se encontram;
- "ArqFinal.o" poderá ser substituído pelo nome que quiser, porém a extensão ".o" deverá ser sempre mantida (Exemplo: "MinhaApp.o");
- Script1.lua, Script2.lua, e Script3.lua deverão ser substituídos pelos seu scripts;
- O sinal "^" no final da linha significa que existirá na próxima linha um novo script.lua a ser inserido na compilação.

Após fazer as alterações salve e execute o "lua.bat", é importante que o computador onde a compilação será gerada tenha um sistema operacional de 32 bits.

Para chamar as funções desses módulos use o comando dofile, porém substitua o módulo(scriptx.lua) pelo módulo com extensão ".o" gerado na compilação :

Exemplo:

```

<data destination=parser>
  dofile("\ArqFinal.o")
  rodaTeste()
</data>
  
```

### 6.0 MIFARE – leitura e escrita

Apenas para os modelos de POS que tenham esta entrada.

O comando para leitura de dados de cartão Mifare está especificado no capítulo: 3.2.19 (Leitura de Dados do Mifare).

Para efetuar a escrita de dados em um determinado setor do cartão Mifare (contactless), devemos usar o comando CTLSWRITE, cujo conteúdo dos dados fica no corpo do comando, conforme abaixo:

```
<CTLSWRITE KEY=yyyyyyyyyyyy KTYPE=a BLOCK=0 SECTOR=0 BCOUNT=3 RC_NAME=vvv>  
dado  
</ CTLSWRITE >
```

Onde:

dado: é a informação a ser gravada **(em HEXA)**.

Vvv é o nome da variável que receberá o código de retorno do comando que deverá ser '00' se o comando foi bem sucedido.

## 7.0 PAGAMENTO

Este comando é utilizado para efetuar transações de Crédito e Débito, Visa e Mastercard, através de redes de adquirência.

O POS efetuará comunicação a um Servidor Sitef.

Para que o POS com NAVS possa aceitar Visa e Mastercard vários requisitos de segurança precisam ser atendidos (PCI), pois a parte do risco da transação fica com o cliente.

Contate nosso Comercial para maiores detalhes.

```
<PAGAMENTO IPTEF= 184.172.45.130 PORTATEF=4096 CODLOJA=00000000 TIPO=MENU  
VALOR=1000 PAGRET=RESPAG BIN=BINCARTAO NINST=NOMEINST NSU=NSUAUT  
NSUSITEF=NSUSIT NSUCANC= NSUC AUT=CAUT NPAR=NPARCEL MODPAG=TIPOTRANS  
TIPOCAR=TCAR REDEADQ=RADQ”;
```

IPTEF e PORTATEF -> IP (não pode ser dns) e Porta do servidor Sitef.

Se ativar modo SSL (config NAVS), estas 2 informações serão ignoradas, pois será usado IP e Porta do certificado SSL, mas devem estar no comando.

CODLOJA -> código da loja, para nosso servidor de teste deve ser 00000000.

A identificação do terminal é feita pelo parâmetro IDTERM que por padrão assumimos no NAVS o número serial do POS menos o primeiro dígito, por exemplo, se o serial do POS é 123-456-789 o IDTERM fica 23456789 automaticamente. Deixar o IDTERM automático (preenchido automaticamente pelo NAVS com o serial do POS) é muito importante pois possibilita o rastreio de transações vinculadas ao POS (número físico do POS) e evita que dois POS's estejam com o mesmo número junto ao Sitef o que pode ocasionar des controle de NSU e erro de transação.

em último caso ( **DESACONSELHÁVEL - RECOMENDAMOS FORTEMENTE NÃO USAR DESSA FORMA**), Caso deseje controlar a identificação do terminal, use o parâmetro:

IDTERM -> identificação do terminal: 8 dígitos e **SEMPRE USAR IDENTIFICAÇÕES DIFERENTES PARA CADA POS** com o mesmo CODLOJA.

A partir da versão 1.34.3, este parâmetro foi desabilitado !!!!!

TIPO -> tipo da transação, as opções aqui são: MENU (habilita um menu com as transações disponíveis), CREDITO, DEBITO, ESTORNO, GERENCIAL, CARTAO (para entrada automática do cartão).

Ou, o código da transação, vide tabela de códigos de funções abaixo (principais).

VALOR -> informar o valor da transação, considerando os centavos (1000 = R\$10,00).

NOPRINT -> opcional, informa se o pagamento emite ou não os recibos (loja e cliente) – 1 ou S não imprime

### A partir da versão 1.18.2:

Variável NPAR, pode ser usada como entrada, para pré-definir o número de parcelas no Crédito Parcelado. Quando colocado um valor numérico, e TIPO=CREDITO, a venda parcelada é automaticamente chamada.

### A partir da versão 1.18.4:

## Especificação de Interface

---

Parâmetro TIPO=113 (reimpressão específica), quando usado junto com a variável REDEADQ, valores 1 (Rede), 2 (Cielo ) ou 3 (Outros) efetua a reimpressão específica diretamente na rede adquirente, evitando que apareça o menu para escolha da rede.

### A partir da versão 1.19.4:

NOPRINT -> valor C ou c, não imprime a via do Cliente; e L ou l, não imprime a via da Loja.

Variáveis de saída (opcionais):

PAGRET -> variável que receberá o resultado da transação: "APROVADO" ou "NAO APROVADO: xxx", xxx é o código de erro, vide tabela de códigos de retorno abaixo (a partir da versão 1.18.5).

BIN -> retorna o BIN do cartão utilizado.

NINST -> nome da instituição.

NSU -> NSU da transação.

NSUSITEF -> NSU do SITEF (a partir da versão 1.18.7).

NSUCANC -> NSU da transação Cancelada (a partir da versão 1.18.7).

AUT -> código de autorização.

NPAR -> número de parcelas.

MODPAG -> tipo da transação.

TIPOCAR -> código do tipo do cartão (código da bandeira).

REDEADQ -> código da rede adquirente.

VCANCEL -> valor do cancelamento com a vírgula decimal e os centavos

RECLOJA -> a via do recibo da loja em hexadecimal no formato de impressão do NAVS (só converter para ASCII e incluir seu conteúdo entre os comandos <PRINTER> e </PRINTER> )

RECCLI -> a via do recibo do cliente em hexadecimal no formato de impressão do NAVS (só converter para ASCII e incluir seu conteúdo entre os comandos <PRINTER> e </PRINTER> )

### A partir da versão 1.30.8:

HDRREC -> Para incluir header nos recibos do Pagamento.

FOOTREC -> Para incluir footer nos recibos do Pagamento.

CUPOMF -> número sequencial do cupom fiscal, controlado pelo cliente. Tem que ser diferente a cada transação.

CNPJCRED -> CNPJ da credenciadora.

FANTASIA -> nome fantasia para impressão no header do pagamento.

ENDERECO -> endereço para impressão no header do pagamento.

CIDADE -> cidade para impressão no header do pagamento.

CNPJ -> cnpj para impressão no header do pagamento.

### A partir da versão 1.31.0:

CREDLOJ -> Crédito Parcelado Loja.

CREDADM -> Crédito Parcelado Administradora.

### A partir da versão 1.35.0:

VPAGO -> Retorna o valor total da transação, ou valor do boleto bancário para CB.

FISCAL -> Retorna string com número do cupom, data e hora fiscal, separados por "-".

DATAANC -> Entrada da data do cancelamento para o ESTORNO.

CUPOMF -> Entrada do cupom fiscal para o ESTORNO.

VALOR-> Entrada do valor para o ESTORNO.

## Especificação de Interface

### Correspondente Bancário (CB):

VPAGO -> Retorna o valor do boleto bancário.

DATAV -> Retorno da data de vencimento do boleto.

FORMAPAG -> Retorna a forma de pagamento.

CB -> Entrada com o Código de barras.

### Obs.:

A consulta tem um parâmetro de retorno: FORMAPAG=FPAG, que retorna o seguinte conteúdo:

"1:1;2:0;3:0;4:0", onde o número a esquerda dos 2 pontos(:) significa:

TIPO	Descrição
1	Dinheiro
2	Cheque
3	TEF Crédito
4	TEF Débito

O conteúdo após os 2 pontos, significa a permissão para o pagamento: 1 – Permite ou 0 – Não Permite

**ATENÇÃO:** Há o limite no NAVS de 20 parâmetros por comando.

Os últimos recibos de pagamento são guardados nos arquivos CVIALOJ (cupom lojista) e CVIACLI (cliente) no POS.

Arquivo de nome PAGAM.RET é gravado no POS, com as informações da transação.

### Tabela de códigos de funções (parâmetro TIPO):

Função	Descrição
0	Pagamento genérico. A CliSiTef permite que o operador escolha a forma de pagamento através de menus.
1	Cheque
2	Débito
3	Crédito
4	Fininvest
5	Cartão Benefício
6	Crédito Centralizado
7	Cartão Combustível
15	Venda com cartão Gift
16	Débito para pagamento de carnê
110	Abre o menu de transações Gerenciais
111	Teste de comunicação com o SiTef
112	Menu Re-impressão
113	Re-impressão comprovante específico
114	Re-impressão ultimo comprovante
115	Pré-autorização
116	Captura de pré-autorização
117	Ajuste de pré-autorização

## Especificação de Interface

118	Consulta de pré-autorização
200	Cancelamento Normal: Inicia a coleta dos dados no ponto necessário para fazer o cancelamento de uma transação de débito ou crédito, sem ser necessário passar antes pelo menu de transações administrativas
201	Cancelamento Telemarketing: Similar a modalidade 200 só que para a função de cancelamento de transação de crédito digitado
202	Cancelamento Pré-autorização
203	Cancelamento da Captura da Pré-autorização
210	Cancelamento de venda com cartão de Crédito
211	Cancelamento de venda com cartão de Débito
212	Cancelamento de venda com cartão Combustível
213	Cancelamento de Venda com Cartão Gift
250	Cancelamento de Consulta Bônus
251	Cancelamento Recarga Cartão Presente
253	Cancelamento Acúmulo de Pontos Cartão Bônus
254	Resgate de Pontos Cartão Bônus
255	Cancelamento de Resgate de Pontos Cartão Bônus
256	Acúmulo de Pontos Cartão Bônus
257	Cancelamento Recarga Cartão Gift
300	Recarga de pré-pago
301	Recarga de pré-pago desvinculada do pagamento
302	Recarga de pré-pago sem coleta de valor
303	Recarga de pré-pago desvinculada do pagamento e sem coleta de valor
310	Corresponde Bancário (Pagamento de Contas)
311	Pagamento de Contas com Saque
600	Consulta Saldo
601	Consulta Saldo Cartão de Débito
602	Consulta Saldo Cartão de Crédito

Abaixo 2 tabelas com os códigos dos tipos do cartão e rede adquirente:

Código da Bandeira (tipo do cartão)	Descrição
00000	Outro, não definido
00001	Visa
00002	Mastercard
00003	Diners
00004	American Express
00005	Sollo
00006	Sidecard (Redecard)
00007	Private Label (Redecard)



## Especificação de Interface

00008	Redeshop
00009	Pão de Açúcar
00010	Fininvest (Visanet)
00011	JCB
00012	Hipercard
00013	Aura
00014	Losango
00015	Sorocred
10014	Discovery

### Outros:

20001	Maestro
20002	Visa Electron
00031	Elo – Crédito
20032	Elo - Débito

Código das Redes Autorizadoras	Descrição
00000	Outra, não definida
00001	Tecban
00002	ITAÚ
00003	BRADESCO
00004	Visanet – Especificação 200001
00005	Redecard
00006	Amex
00007	SOLLO
00008	E CAPTURE
00009	Serasa
00010	SPC Brasil
00011	SERASA DETALHADO
00012	TELEDATA
00013	ACSP
00014	ACSP DETALHADO
00015	TECBIZ
00016	CDL DF
00017	Repom
00018	STANDBY
00019	EDMCARD
00020	CREDICESTA
00021	Banrisul
00022	ACC CARD
00023	Clubcard
00024	ACPR
00025	Vidalink
00026	CCC_WEB
00027	Ediguay
00028	Carrefour
00029	Softway
00030	Multicheque
00031	Ticket combustível
00032	YAMADA
00033	Citibank
00034	Infocard
00035	BESC
00036	ISSO
00037	CHEQUE CASH
00038	Central Card
00039	Drogaraia
00040	OUTRO SERVIÇO
00041	Accor
00042	EPAY GIFT
00043	Parati
00044	TOKORO
00045	Coopercred
00046	SERVCEL
00047	Sorocred

## Especificação de Interface

00048	Vital
00049	SAX FINANCEIRA
00050	Formosa
00051	Hipercard
00052	Tricard
00053	CHECK OK
00054	Policard
00055	Cetelem Carrefour
00056	LEADER
00057	Consórcio Credicard Venezuela
00058	GAZINCRED
00059	Telenet
00060	Cheque Pré
00061	Brasil Card
00062	Epharma
00063	Total
00064	Consórcio Amex Venezuela
00065	GAX
00066	Peralta
00067	SERVIDOR PAGAMENTO
00068	BANESE
00069	RESOMAQ
00070	Sysdata
00071	CDL POA
00072	BIGCARD
00073	DTRANSFER
00074	VIAVAREJO
00075	CHECK EXPRESS
00076	Givex
00077	Valecard
00078	Portal Card
00079	Banpara
00080	SOFTNEX
00081	SUPERCARD
00082	GetNet
00083	Prevsaude
00084	BANCO POTTENCIAL
00085	SOPHUS
00086	MARISA 2
00087	MAXICRED
00088	BLACKHAWK
00089	EXPANSIVA
00090	SAS NT
00091	LEADER 2
00092	SOMAR
00093	CETELEM AURA
00094	CABAL
00095	CREDSYSTEM
00096	Banco Provincial
00097	CARTESYS
00098	CISA
00099	TRNCENTRE
00100	ACPR D
00101	CARDCO
00102	CHECK CHECK
00103	CADASA
00104	PRIVATE BRADESCO
00105	CREDMAIS
00106	GWCEL
00107	CHECK EXPRESS 2
00108	GETNET PBM
00109	USECRED
00110	SERV VOUCHER
00111	TREDENEXX
00112	Bonus Presente Carrefour
00113	CREDISHOP
00114	ESTAPAR

## Especificação de Interface

00115	Banco Ibi
00116	WORKERCARD
00117	Telecheque
00118	OBOE
00119	PROTEGE
00120	SERASA CARDS
00121	Hotcard
00122	Banco Panamericano
00123	Banco Mercantil
00124	Sigacred
00125	Visanet – Especificação 4.1
00126	SPTRANS
00127	PRESENTE MARISA
00128	COOPLIFE
00129	BOD
00130	G CARD
00131	TCREDIT
00132	SISCRED
00133	FOXWINCARDS
00134	CONVCARD
00135	Voucher
00136	EXPAND CARDS
00137	Ultragaz
00138	Qualicard
00139	HSBC UK
00140	Wappa
00141	SQCF
00142	INTELLISYS
00143	BOD DÉBITO
00144	ACCREDITO
00145	COMPROCARD
00146	ORGCARD
00147	MINASCRED
00148	Farmácia Popular
00149	Fidelidade Mais
00150	ITAÚ SHOPLINE
00151	CDL RIO
00152	FORTCARD
00153	PAGGO
00154	SMARTNET
00155	INTERFARMACIA
00156	VALECON
00157	CARTÃO EVANGÉLICO
00158	VEGASCARD
00159	SCCARD
00160	ORBITALL
00161	ICARDS
00162	FACILCARD
00163	FIDELIZE
00164	FINAMAX
00165	BANCO GE
00166	UNIK
00167	TIVIT
00168	VALIDATA
00169	BANESCARD
00170	CSU CARREFOUR
00171	VALESHOP
00172	SOMAR CARD
00173	OMNION
00174	CONDOR
00175	STANDBYDUP
00176	BPAG BOLDCRON
00177	MARISA SAX SYSIN
00178	STARFICHE
00179	ACE SEGUROS
00180	TOP CARD
00181	GETNET LAC

## Especificação de Interface

00182	UP SIGHT
00183	MAR
00184	FUNCIONAL CARD
00185	PHARMA SYSTEM
00186	NEUS
00187	SICREDI
00188	ESCALENA
00189	N SERVIÇOS
00190	CSF CARREFOUR
00191	ATP
00192	AVST
00193	ALGORIX
00194	AMEX EMV
00195	COMPREMAX
00196	LIBERCARD
00197	SEICON
00198	SERASA AUTORIZ CRÉDITO
00199	SMARTN
00200	PLATCO
00201	SMARTNET EMV
00202	PROSA MÉXICO
00203	PEELA
00204	NUTRIK
00205	GOLDENFARMA PBM
00206	GLOBAL PAYMENTS
00207	ELAVON
00208	CTF
00209	BANESTIK
00210	VISA ARG
00211	AMEX ARG
00212	POSNET ARG
00213	AMEX MÉXICO
00214	ELETROZEMA
00215	BARIGUI
00216	SIMEC
00217	SGF
00218	HUG
00219	CARTÃO CONSIGNUM CARTÃO METTACARD
00220	DDTOTAL
00221	CARTÃO QUALIDADE
00222	REDECONV
00223	NUTRICARD
00224	DOTZ
00225	PREMIAÇÕES RAIZEN
00226	TROCO SOLIDÁRIO
00227	AMBEV SÓCIO TORCEDOR
00228	SEMPRE
00229	FIRST DATA
00230	COCIPA
00231	IBI MÉXICO
00232	SIANET
00233	SGCARDS
00234	CIAGROUP
00235	FILLIP
00236	CONDUCTOR
00237	LTM RAIZEN
00238	INCOMM
00239	VISA PASS FIRST
00240	CENCOSUD
00241	HIPERLIFE
00242	SITPOS
00243	AGT
00244	MIRA
00245	AMBEV 2 SÓCIO TORCEDOR
00246	JGV
00247	CREDSAT
00248	BRAZILIAN CARD

## Especificação de Interface

00249	RIACHUELO
00250	ITS RAIZEN
00251	SIMCRED
00252	BANCRED CARD
00253	CONEKTA
00254	SOFTCARD
00255	ECOPAG
00256	C&A AUTOMAÇÃO IBI
00257	C&A PARCERIAS BRADESCARD
00258	OGLOBA
00259	BANESE VOUCHER
00260	RAPP
00261	Monitora POS
00262	SOLLUS
00263	FITCARD
00264	ADIANTI
00265	STONE
00266	DMCARD

### IMPORTANTE:

Apesar do terminal não conter Masterkey, será possível efetuar transações de Crédito sem senha, e Crédito com senha usando cartão com chip.

Porém, **cuidado** com a senha usada em cartão com chip, ela deve ser a senha real do cartão, pois ela é validada Offline e caso erre a senha mais de 2 vezes o cartão será **BLOQUEADO**.

Através do arquivo CLSIT, é possível habilitar ou desabilitar transações. Abaixo um exemplo de uso deste arquivo.

Usar na sessão chamada [Geral]:

[Geral]

TransacoesHabilitadas=16;26;30

TransacoesDesabilitadas=29;36;42;58

Este arquivo fica na pasta "1" da carga do NAVS.

Abaixo parte da tabela dos códigos mais importantes:

Cartão de débito a vista	16
Cartão de débito pré-datado	17
Cartão de débito parcelado	18
Cartão de débito CDC	19
Cartão Refeição Eletrônico (Exceto Visanet, neste caso usar 3014)	20
Voucher Papel	21
Cartão Benefício	22
Cartão SmartVR/Sodexo com leitora SmartNet	23
Cartão de crédito a vista com juros	24
Cartão de crédito (todas as combinações) (Descontinuado, não usar)	25
Cartão de crédito a vista	26
Cartão de crédito parcelado com financiamento do estabelecimento	27
Cartão de crédito parcelado com financiamento da administradora	28

## Especificação de Interface

Cartão de crédito digitado	29
Cartão de crédito magnético	30
Pré-autorização	31
Consulta parcelas no Cartão de Crédito	36
Crédito Cisa	37
Saldo/Extrato Cisa	38
Cartão Crédito Infocard	39
Cancelamento de transação com cartão de crédito ou débito	40
Consulta AVS	41
Débito Digitado	42
Débito Magnético	43
Crédito Parcelado	44
Reimpressão	56
Reimpressão do Último Comprovante	57
Reimpressão Específica	58

### Tabela de códigos de retorno

Valor	Descrição
0	Sucesso na execução da função.
10000	Deve ser chamada a rotina de continuidade do processo.
outro valor positivo	Negada pelo autorizador.
-1	Módulo não inicializado. O PDV tentou chamar alguma rotina sem antes executar a função configura.
-2	Operação cancelada pelo operador.
-3	O parâmetro função / modalidade é inválido.
-4	Falta de memória no PDV.
-5	Sem comunicação com o SiTef.
-6	Operação cancelada pelo usuário (no pinpad).
-7	Reservado
-8	A CliSiTef não possui a implementação da função necessária, provavelmente está desatualizada (a CliSiTefI é mais recente).
-9	A automação chamou a rotina ContinuaFuncaoSiTefInterativo sem antes iniciar uma função iterativa.
-10	Algum parâmetro obrigatório não foi passado pela automação comercial.
-12	Erro na execução da rotina iterativa. Provavelmente o processo iterativo anterior não foi finalizado até o final (enquanto o retorno for igual a 10000).
-15	Operação cancelada pela automação comercial.
-20	Parâmetro inválido passado para a função.
-21	Utilizada uma palavra proibida, por exemplo SENHA, para coletar dados em aberto no pinpad.

	Por exemplo na função ObtemDadoPinpadDiretoEx.
-25	Erro no Correspondente bancário: Deve realizar sangria.
-30	Erro de acesso ao arquivo. Certifique-se que o usuário que roda a aplicação tem direitos de leitura/escrita.
-40	Transação negada pelo SiTef.
-41	Dados inválidos.
-42	Reservado
-43	Problema na execução de alguma das rotinas no pinpad.
-50	Transação não segura.
-100	Erro interno do módulo.
outro valor negativo	Erros detectados internamente pela rotina.

## 8.0 WEB SERVICES

### Comando WS\_GET e WS\_POST

**<WS\_GET** HOST=*www.ws.com* PATH=*/dir/ws* IP=*123.456.789* PORTA=*1234* Header1=*value1* Header2=*value2* ADD=*Header1,Header2* NAME=*RET*>

**<WS\_POST** HOST=*www.ws.com* PATH=*/dir/ws* IP=*123.456.789* PORTA=*1234* Header1=*value1* Header2=*value2* ADD=*Header1,Header2* NAME=*RET* >

*conteúdo de BODY da requisição HTTP*

**</WS\_POST>**

### Parâmetros:

**HOST:** Especifica o host de acesso para o Webservice (**SE ESTE PARÂMETRO FOR DEFINIDO, NÃO ESPECIFICAR O IP**).

**IP:** Especifica o IP de acesso para o Webservice (**SE ESTE PARÂMETRO FOR DEFINIDO, NÃO ESPECIFICAR O HOST**).

**PORTA:** Especifica a porta de acesso para o Webservice. Se este parâmetro não for especificado, será considerado a porta padrão (HTTP-80, HTTPS-443).

### ADICIONANDO CAMPOS ADICIONAIS NA REQUISIÇÃO HTTP:

Os comandos WS\_POST e WS\_GET permitem que o usuário possa definir campos adicionais no header da requisição HTTP.

Para fazer isso, basta incluir os nomes dos campos no parâmetro ADD e depois especificar seus valores utilizando a forma PARAMETRO=VALOR.

Veja um exemplo de como adicionar dois campos adicionais (Authorization e Content-Type) na requisição HTTP:

**<WS\_GET Authorization=12345 Content-Type=application/json ADD=Authorization,Content-Type...**

Observe que o parâmetro ADD deve ser definido por último.



## 9.0 CRIPTOGRAFIA

### Comando CRYPT

```
<CRYPT MODEL=AES FUNC=1 KEY=12345 IV=12345 NAME=RET>  
      conteúdo a ser criptografado  
</CRYPT>
```

#### **Parâmetros:**

Model: Especifica o tipo de criptografia a ser utilizado. Nesta versão, é aceito o tipo AES256 e DESNBS.

FUNC: Especifica se o conteúdo será criptografado ou descriptografado.

KEY: Especifica a chave de criptografia.

IV: Especifica a IV de criptografia (somente para AES-256).

NAME: Especifica o nome da variável de retorno.

## 10.0 GPS

A partir da versão 1.36.0, o NAVS consegue obter as coordenadas de latitude e longitude através de uma máquina de estado que é executada paralelamente com o NAVS. Observe que, essa máquina de estado nunca para de ser executada – ela é executada de tempos e tempos até que o POS seja desligado ou se a variável (**NGPS**) que ativa esse modo estiver configurada como “0”.

O comando <GPS> captura somente **A ÚLTIMA** informação obtida pela máquina de estado. Ou seja, a informação obtida pelo comando pode não corresponder ao momento exato em que o comando foi executado. É possível obter também a data e a hora de quando as coordenadas foram capturadas.

Exemplo de um comando GPS:

```
<GPS LAST_READ=LR LAST_SUCCESS=LS GPS_ERROR=GPSE>
```

Basicamente, o comando captura três tipos informações:

### 1. LAST\_READ

A última tentativa de leitura da máquina de estado.

### 2. LAST\_SUCCESS

A última vez que em que a máquina de estado conseguir obter as coordenadas de latitude e longitude com sucesso.

### 3. GPS\_ERROR

Especifica se a máquina de estado conseguiu ou não obter as coordenadas de latitude e longitude.

O parâmetro LAST\_READ retorna a informação no seguinte formato:

**MSG:xxx,TS:yyyymmddhhmmssw**

**MSG:** Retorna o código de erro obtido pela última tentativa de leitura.

**TS:** Retorna a data e a hora da última tentativa de leitura, onde YYYY é o ano, MM é o mês, DD é o dia, HH é a hora, MM é o minuto, SS é o segundo e o W é dia da semana de 0 a 6.

O parâmetro LAST\_SUCCESS retorna a informação no seguinte formato:

**LT:00.000000,LN:00.000000,TS:yyyymmddhhmmssw**

**LT:** Retorna a coordenada de latitude

**LN:** Retorna a coordenada de longitude

**TS:** Retorna a data e a hora da última vez que obteve as coordenadas, onde YYYY é o ano, MM é o mês, DD é o dia, HH é a hora, MM é o minuto, SS é o segundo e o W é dia da semana de 0 a 6.

## Especificação de Interface

---

Para ativar a máquina de estado de captura do GPS, basta configurar as seguintes variáveis de ambiente:

### NGPS

- 1 – Ativa a máquina de estado de captura.
- 0 – Desativa a máquina de estado de captura.

### NGPSTIME

Especifica o tempo de intervalo (minutos) em que a captura é executada. O valor mínimo é de 3 minutos.

A seguir, as descrições do código de erro:

- 0** Operação de leitura realizada sem erros.
- 100** A variável de ambiente NGPS está configurada como 0.
- 101** A máquina de estados não conseguiu obter as coordenadas.
- 200** Erro de inicialização (GPS\_ENGINE).
- 201** Erro de inicialização (GPS\_OPEN).
- 202** Erro de inicialização (GPS\_STREAM).
- 203** Erro na leitura de coordenadas (GPS\_READ).
- 204** Erro na leitura de coordenadas (SOCKET).
- 205** A máquina de estado foi interrompida no meio da operação.

## **11.0 Descrição da Mensagem do POS ao Servidor**

Todos os campos capturados serão enviados para o Servidor quando for encontrado o comando POST, da seguinte forma:

nome1=campo1&nome2=campo2&...

Onde, “nome” é o argumento coletado no parâmetro NAME do campo em questão, e “campo” é o valor capturado pelo POS.

A ordem na qual os campos e seus respectivos valores aparecem na string de mensagem é a ordem em que o comando de captura dos respectivos campos aparecem no corpo do XML.

## 12.0 Exemplo de uso

Como exemplo, a sequência da comunicação entre POS e Servidor poderá ser:

- 1) POS inicia comunicação com Servidor baseado no arquivo XML previamente carregado no POS.
- 2) Servidor responde com os comandos para iniciar a transação no POS:

```
<INIT>
<CONSOLE>-----VENDA-----<BR> PASSE O CARTÃO<BR><\CONSOLE>
<GET NAME=CARTAO TYPE=CARD TRACKS=3>
<CONSOLE> DIGITE O VALOR<BR><\CONSOLE>
<GET NAME=VALOR TYPE=FIELD SIZE=5 COL=8 LIN=2>
<POST>
```

- 3) POS mostrará no display:

```
"-----VENDA-----"
" PASSE O CARTAO"
```

O POS, então, aguardará a passagem do cartão pelo usuário e irá para o próximo passo da transação:

```
" DIGITE O VALOR "
```

Ao iniciar a digitação, teremos no display do POS:

```
" DIGITE O VALOR "
23
```

Após apertar o Enter no teclado, o POS enviará ao Servidor o texto abaixo:

```
CARTAO="trilhas 1 e 2"&VALOR=23
```

- 4) Servidor poderá responder ao POS:

```
<CONSOLE>CARTAO ACEITO<BR><\CONSOLE>
<CONSOLE>QUANTIDADE: 23<BR><\CONSOLE>
<PRINTER>CARTAO: XXXXXXXXXXXX</PRINTER >
<INIT>
<CONSOLE>-----VENDA-----<BR> PASSE O CARTÃO<BR><\CONSOLE>
<GET NAME=CARTAO TYPE=CARD TRACKS=3>
```

## Especificação de Interface

---

<CONSOLE> DIGITE O VALOR<BR><\CONSOLE>  
<GET NAME=VALOR TYPE=FIELD SIZE=5 COL=8 LIN=2>  
<POST>

**5)** O POS mostrará no display:

“CARTAO ACEITO”  
“QUANTIDADE: 23”

E, imprimirá:

“CARTAO: XXXXXXXXXXXX”

**6)** O POS ficará pronto para a próxima transação, mostrando no display:

“-----VENDA-----”  
“ PASSE O CARTAO”

**7)** O POS sempre voltará para este ponto sempre que a tecla CANCELA for acionada:

“-----VENDA-----”  
“ PASSE O CARTAO”

## 13.0 Configuração do NAVS

Para fazer as configurações no NAVS relativas à Conexão, Rede e Servidor aperte a tecla '#' durante aproximadamente 5 segundos, e siga os passos abaixo:

Primeiramente é solicitado a digitação do código de acesso, controlado por uma variável do sistema operacional.

A variável é \*OPP.

Caso esta variável não exista, basta acionar Enter após a solicitação do código.

### CONFIGURAÇÃO

**1 CONEXAO** = Escolha do tipo de comunicação, opte por: **1 GPRS**, **2 ETHERNET**, **3 WIFI**.

A partir da versão 1.19.1:

1 SO GPRS, 2 SO ETHERNET, 3 GPRS+ETHERNET, 4 ETHERNET+GPRS (para modelos com GPRS e ETH).

1 SO WIFI, 2 SO GPRS, 3 WIFI+GPRS, 4 GPRS+WIFI (para modelos com WIFI e GPRS).

E, quando escolher 2 tipos de comunicação, terá nova tela para a escolha da configuração de cada conexão, e o NAVS automaticamente irá mudar o tipo quando houver queda da outra comunicação.

**2 REDE** = Configure de acordo com a opção escolhida em CONEXÃO:

a) Para modelos que não suportam Dual Chip:

Configura o SIM 1, conforme abaixo:

1 MANUAL: para digitar Operadora do Chip manualmente.

2 a 7: escolha a operadora: Vivo, Claro, Oi, Tim, Lyra, Vodafone, Avatek, Datelo ou VectoMobile e a APN, Usuário e Senha serão atualizados automaticamente.

b) Para modelos que suportam Dual Chip, exemplo: Vx690:

- Se escolheu **1 GPRS**:

1 CONFIG SIMS

Configura o modo de operação:

1 UNO SIM1: vai usar apenas o SIM 1

## Especificação de Interface

---

- 2 UNO SIM2: vai usar apenas o SIM 2
- 3 DUAL MANUAL: vai usar o modo manual somente controlado pelo comando GPRS\_SWITCH\_SIM
- 4 DUAL AUTOMATICO: vai usar o modo automático, onde automaticamente chaveará de SIM caso caia a conexão do outro SIM.

### 2 PARAM SIM1

Configura o SIM 1, conforme abaixo:

- 1 MANUAL: para digitar Operadora do Chip manualmente.
- 2 a 7: escolha a operadora: Vivo, Claro, Oi, Tim, Lyra, Vodafone, Datelo, Avatek, VectoMobile, e a APN, Usuário e Senha serão atualizados automaticamente.

### 3 PARAM SIM2

Configura o SIM 2, conforme abaixo:

- 1 MANUAL: para digitar Operadora do Chip manualmente.
- 2 a 7: escolha a operadora: Vivo, Claro, Oi, Tim, Lyra, Vodafone, Datelo, Avatek, VectoMobile, e a APN, Usuário e Senha serão atualizados automaticamente.

4 2G ou 3G: com as opções: DEFAULT, AUTOMATICO, SOMENTE 2G, SOMENTE 3G

5 Editar Ping: editar o intervalo do Ping: Definir o tempo em minutos (padrão 15 min.) para envio do ping para manter a rede conectada. Para não enviar o ping colocar o valor 0. Obs.: são enviados 56 bytes.

6. DESATIVAR GPRS: desativa GPRS caso simcard ausente.

**APN:** Digitar a APN da operadora. Ex.: claro.com.br/ gprs.oi.com.br/ tim.br/ zap.vivo.com.br – Todas estas APN publicas.

**USUARIO:** Sempre o nome da operadora. Ex.: claro / oi / tim / vivo.

**SENHA:** Sempre o nome da operadora. Ex.: claro / oi / tim / vivo.

- Se escolheu **2 ETHERNET:**

#### 1 IP ESTATICO

#### 2 IP DINAMICO

- Se escolheu 1 IP ESTATICO:

**IP:** Digite o IP do ponto de rede que está conectado no POS.



## Especificação de Interface

---

**MASCARA:** Digite a MASCARA do ponto de rede que está conectado no POS.

**GATEWAY:** Digite a GATEWAY do ponto de rede que está conectado no POS.

**DNS1:** Digite o DNS1 do ponto de rede que está conectado no POS.

**DNS2:** Digite o DNS2 do ponto de rede que está conectado no POS.

Caso utilize o modo DHCP, atualize os campos acima com: **0.0.0.0**

**PING:** Definir o tempo em minutos para envio do ping para manter a rede conectada. Para não enviar o ping colocar o valor 0.

- Se escolheu 2 IP DINAMICO:

IP, MASCARA, GATEWAY DNS1 E DNS2 DEFINIDOS COMO: DINAMICOS TECLE ENTER PARA RETORNAR.

- Se escolheu 3 WIFI:

**1 SSID:** Digite o SSID da rede Wi-Fi que está conectado no POS.

**2 SENHA:** Digite a senha da rede Wi-Fi que está conectado no POS.

**3 SEGURANCA WIFI:** escolha o modo de segurança. WPA-PSK ou WPA2-PSK, e depois o algoritmo WPA (TKIP ou AES CCM).

**4 CANAL:** digite o canal.

**5 IP ESTATICO:** Digite o IP da rede Wi-Fi que está conectado no POS.

**MASCARA:** Digite a MASCARA da rede Wi-Fi que está conectado no POS.

**GATEWAY:** Digite a GATEWAY da rede Wi-Fi que está conectado no POS.

**DNS1:** Digite o DNS1 da rede Wi-Fi que está conectado no POS.

**DNS2:** Digite o DNS2 da rede Wi-Fi que está conectado no POS.

**6 IP DINAMICO:** Selecione esta opção para o modo DHCP.

**7 PING:** Definir o tempo em minutos para envio do ping para manter a rede conectada. **Usar o valor 1 para conexão WiFi para evitar a queda com o roteador.**

**8 LISTA DE REDES WIFI:**

0-Desativar: vale o que está configurado e não exibe a lista.

## Especificação de Interface

---

- 1-Exibir sempre: vai exibir a lista das redes sempre que o POS foi iniciado.
- 2-Exibir somente em erros: vai exibir a lista somente em casos de erros de conexão.
- 3-Exibir apenas uma vez: vai exibir a lista somente 1 vez.
- 4-Exibir apenas uma vez (\*): a lista é exibida apenas 1 vez, ou sempre que ocorrer erro de conexão.

### 9 RECONEXAO PONTO DE ACESSO:

- 1-Limite Minimo de Sinal , se definido com valor maior que 0, haverá reconexão automática em queda de sinal, para redes com mesmo SSID.
- 2- Tempo Reconexao
- 3- Exibicao Barra de Status, se valor =1, será exibido os valores no lugar do nome da rede na linha de status.

#### OBS.:

Para uso em rede Wi-Fi invisível (sem broadcast), no modo WPA e WPA2, configure o SSID, SENHA e SEGURANÇA WIFI.

### 3 SERVIDOR

**IP DESTINO:** Digite o IP ou o endereço DNS do Servidor que responderá às solicitações do POS (sem o www). Exemplo: google.com.br.

Obs.: É importante ressaltar que o DNS permita conversão para um ip correspondente, ou seja, não é possível hospedar uma aplicação NAVS num servidor de IP compartilhado.

**PORTA:** Digite a PORTA do Servidor que responderá às solicitações do POS.

**RECURSO:** Digite o nome da aplicação do Servidor.

**HOST:** Digite o nome do Host, pode ficar em branco na maioria dos casos, mas se tiver algum problema relacionado á DNS, **talvez**, preenchendo este campo o problema seja solucionado.

### HTTPS: Ativa Menu HTTPS

HTTPS.1) Ativa/Desativa

HTTPS.2) Certif Local/Server

HTTPS.3) Metodo HTTPS (escolha entre SSLv3, TLSv1, TLSv1.1 e TLSv1.2), **para certificado local somente o método TLS irá funcionar**).

HTTPS.4) Nomes Certificados

- 1) Certificado
- 2) Private Key
- 3) Certif CA

HTTPS.5) SNI

- 1) Usar o HOST (usa o valor do parâmetro HOST)
- 2) Desativar

## 4 DIVERSOS

- 1 LINHA STATUS
- 2 SERIAL
- 3 SCROLL
- 4 IMPR. / ECONOMIA.
- 5 PAGAMENTO
- 6 KEEP ALIVE

### 4.1) LINHA STATUS

- 1 POSIÇÃO
- 2 HORA
- 3 BATERIA

#### 4.1.1) POSIÇÃO LINHA DE STATUS

**1 TOPO:** As mensagens de status do POS ficarão na primeira linha da console.

**2 RODAPÉ:** As mensagens de status do POS ficarão na última linha da console.

**3 AUSENTE:** As mensagens de status do POS não aparecerão na console.

#### 4.1.2) HORA LINHA STATUS

- 1 **MOSTRAR:** Mostra hora na linha de status.
- 2 **OCULTAR:** Não mostra hora na linha de status.

#### 4.1.3) STATUS BATERIA

- 1 **FORMA EXIBIÇÃO:** Percentual ou Ícone
- 2 **LIMITE DE ALARME:** Percentual de carga da bateria em que o Ícone de alarme aparece

#### 4.2) SERIAL

Configure os parâmetros da comunicação Serial: Baudrate, Paridade, Data Bits, Stop Bits e Timeout (SER-TIMEOUT em milissegundos).

OBS.: Caso a opção TIMEOUT esteja vazia ou com o valor '0', não haverá o controle de Timeout.

#### 4.3) SCROLL

Configure as teclas para efetuar o scroll de linhas, teclas de subida e descida. Exclusiva para o Vx520.

- 1 **SCROLL UP:** Tecla para efetuar o scroll para cima.
- 1 **SCROLL DOW:** Tecla para efetuar o scroll para baixo.

Selecione a função Scroll (Up ou Dow) e aperte a tecla desejada, no visor aparecerá o código hexa da tecla.

Para finalizar a operação pressione CANCEL.

Não pressione ENTER, senão a tecla Enter será selecionada para o Scroll.

Abaixo alguns exemplos de caracteres hexadecimal das teclas:

TECLA	VALOR HEXA
F1	7 <sup>a</sup>

<b>F2</b>	7B
<b>F3</b>	7C
<b>F4</b>	7D
<b>Roxa 1</b>	61
<b>Roxa 2</b>	62
<b>Roxa 3</b>	63
<b>Roxa 4</b>	64

#### **4.4) IMPR. / ECONOMIA**

##### **4.4.1) CONTRASTE IMPRESS.**

Configure o nível de contraste para a impressão.

**NÍVEL 1:** Configura para o nível 1, padrão da impressora.

**NÍVEL 2:** Configura para o nível 2.

**NÍVEL 3:** Configura para o nível 3. Até aqui para os terminais VX685 e VX690

**NÍVEL 4:** Configura para o nível 4.

**NÍVEL 5:** Configura para o nível 5.

**NÍVEL 6:** Configura para o nível 6.

**NÍVEL 7:** Configura para o nível 7.

**NÍVEL 8:** Configura para o nível 8, nível máximo.

##### **4.4.2) NÍVEL BACKLIGHT TELA**

Configure o nível de iluminação do LCD de 0 (desligado) a 100 (máxima)

##### **4.4.3) NÍVEL BACKLIGHT TECLADO**

Configure o nível de iluminação do teclado de 0 (desligado) a 100 (máxima)

##### **4.4.4) ECONOMIA DE ENERGIA**

Configura se as rotinas internas de economia de energia estão ativas ou não. Ver capítulo a parte.

##### **4.4.5) TEMPO PARA DORMIR**

Configura o tempo em segundos para o POS entrar no modo StandBy (default 40 segs).

### **4.5) PAGAMENTO**

#### **4.5.1) MUDAR STATUS PAGTO**

##### **4.5.1.1) 1 PRESSIONE 1 PARA MUDAR STATUS PGM RELICENCIANDO !!**

Digite 1 para confirmar a mudança de status, onde será solicitado novo licenciamento.

#### **4.5.2) SENHA SUPERVISOR**

Configure a senha para funções de Supervisor.

#### **4.5.3) IMPR. DOS RECIBOS**

Escolha o tipo de impressão dos recibos de pagamento: Normal, Ampliado, Não Imprime e Ponto da Impressao.

##### **4.5.3.1) PONTO DA IMPRESSAO**

1 APOS RETIRAR CARTAO

2 ANTES DE RETIRAR CARTAO

#### **4.5.4) SSL VIA CERT LOCAL**

Ative o modo SSL para certificado no POS.

#### **4.5.5) TRANSACOES PENDENTES**

Desfaz ou Confirma transações pendentes.

Padrão é Desfaz.

##### **4.5.5.1) TRANSACOES PENDENTES**

1 DESFAZ

2 CONFIRMA

#### **4.5.6) LIMPAR TABELAS**

Limpe as tabelas internas do Clisitef, protegido pela senha do Supervisor.

#### **4.5.7) TESTE DE CHAVES**

Imprime e mostra no Display os slots das chaves MasterKey, e nome da rede Adquirente.

#### **4.5.8) FORCA DESCONEXAO**

## Especificação de Interface

---

Força a desconexão com o Sitef no final de cada transação, o padrão é SIM.

1 SIM

2 NAO

### 4.6) KEEP ALIVE

Configure se o número de série do terminal será enviado de tempos em tempos para um endereço e porta previamente definidos.

**ATIVAR / DESATIVAR :** Define se keep alive estará ou não ativado.

**INTERVALO:** Define tempo de intervalo para envio dos pacotes com o número de série.

**IP:** Define o ip de destino, que pode ser diferente do IP do servidor onde está a aplicação.

**PORTA:** Define a porta de destino, que pode ser diferente da Porta do servidor onde está a aplicação.

**GPS:** Ative / Desative o GPS nesta opção.

Obs.: Com o Keep Alive ativo, o POS abre um socket e envia somente o número de série e depois fecha o socket. A frequência é definida de acordo com o Intervalo preenchido pelo usuário.

### 4.7) GPS

Ative / Desative o GPS nesta opção, para os modelos que suportam GPS, exemplo Vx690.

**ATIVAR / DESATIVAR :** Define se o GPS estará ou não ativado.

**TEMPO DE INTERVALO:** Intervalo em minutos para capturar o GPS.

## 5)VERSOES

1 NAVS

SISTEMA OPERACIONAL

2 EOS

4 BC

5 EMV

6 CLISITEF

## **Especificação de Interface**

---

### **7 IMPRIMIR**

#### **5.1) NAVS**

Informa a versão de NAVS em uso no terminal.

#### **5.2) SISTEMA OPERACIONAL**

Informa a versão do sistema operacional em uso no terminal.

#### **5.3) EOS**

Informa a versão de EOS em uso no terminal.

#### **5.4) BC**

Informa a versão de Interface Padrão em uso no terminal.

#### **5.5) EMV**

Informa a versão de EMV em uso no terminal.

#### **5.6) CLISITEF**

Informa a versão de CLISITEF em uso no terminal.

#### **5.7) IMPRIMIR**

Imprime as versões.

### **6) LICENCIAMENTO POS**

- 1 CONFIG. SERVIDOR**
- 2 AVISO**
- 3 INFO**
- 4 SINCRONIZA DATA SERVIDOR**
- 5 RELICENCIA**
- 6 MOSTRA MENSAGENS DO LOL**

#### **6.1) CONFIG. SERVIDOR**

- 1) IP DESTINO**
- 2) PORTA**
- 3) RECURSO**



### 4) HOST

**6.1.1) IP DESTINO:** Digite o IP ou o endereço DNS do Servidor de validação de licença que responderá às solicitações do POS (sem o www). Exemplo: google.com.br.

Obs.: É importante ressaltar que o DNS permita conversão para um ip correspondente, ou seja, não é possível hospedar uma aplicação NAVS num servidor de IP compartilhado.

**6.1.2) PORTA:** Digite a PORTA do Servidor de validação de licença que responderá às solicitações do POS.

**6.1.3) RECURSO:** Digite /LICENCAONLINE/POS.

**6.1.4) HOST:** Digite o nome do Host, pode ficar em branco na maioria dos casos, mas se tiver algum problema relacionado á DNS, **talvez**, preenchendo este campo o problema seja solucionado.

### 6.2) AVISO

Defina a quantidade de dias antes do vencimento da licença online que o NAVS a partir da qual o NAVS mostra que a licença online está pra vencer toda vez que for iniciado.

A tecla vermelha volta para o menu anterior salvando as configurações indicadas.

Caso o valor deste campo seja 0, o POS não mostrará nenhuma mensagem ao usuário relativo ao licença online (LOL).

### 6.3) INFO

Mostra no display as informações da licença online como: número do série licenciado, versão do NAVS, nome do cliente, CNPJ/CPF, data de expiração da licença e data do POS.

### 6.4) SINCRONIZA DATA SERVIDOR

Se SIM, atualiza a data do POS com a data do Servidor do LOL, durante o processo de licenciamento do NAVS.

### 6.5) RELICENCIA

Confirmar com a Tecla 1, se quiser relicenciar.

### 6.5) MOSTRA MENSAGENS DO LOL

Se NAO, todas as mensagens do LOL não serão mostradas e o Licenciamento será automático.

Nova variável criada: NAVS\_MSGS\_LOL, com valores “S” ou “N”.

O padrão é SIM.

### 14.0 Economia de energia

A partir da versão 1.10.5, o NAVS habilita, via configuração, a possibilidade de colocar, automaticamente, o terminal em estado de hibernação após um determinado tempo de ociosidade. Os passos desse processo são os seguintes:

- O terminal, estando ocioso, após 30 segundos emite a seguinte mensagem na linha de status: **“Tecla algo para voltar!!!”** e inicia o processo de hibernação que tem as seguintes fases:
- A iluminação do LCD e do teclado vai diminuindo até desligar completamente;
- Estando desligada a iluminação, o terminal entra em estado de hibernação;
- Para acordá-lo, tecla ENTER;
- Para voltar a trabalhar, após a iluminação voltar, é necessário, conforme a mensagem acima, teclar qualquer tecla;
- Se, após 30 minutos de hibernação, o terminal não for acordado, ele se desligará automaticamente.

Essa nova facilidade permite uma economia considerável da carga da bateria.

Não está disponível para terminais de mesa como o VX520 e VX520C.


## 15.0 Status do Terminal

Na linha de status da console dos terminais, o NAVS apresenta o status do nível de sinal, operadora, nível da bateria, fim de papel e status da comunicação com as seguintes informações:

- Nível do SINAL: 

O nível do sinal GPRS e Wi-Fi é apresentado com as barras indicando o nível.

Caso o sistema operacional não consiga informar o sinal, será apresentado a figura: 

Quando o NAVS estiver configurado com a conexão Ethernet, será apresentada a figura: 

- 3G: indicação quando chip estiver operando em 3G.

- Operadora (GPRS): 

- Nível da BATERIA: 

- A partir da versão 1.18.1, aparecerá indicação (raio) no centro da bateria, quando POS estiver carregando.

O nível da bateria será apresentado apenas para os modelos móveis (ex. Vx680), conforme o nível da bateria, através das figuras acima.

- Nível da bateria em Percentual: **BAT=xxx%**

- Alarme para ligação do POS na tomada de força para recarga da bateria: 

- Status de Fim de Papel: 

- Status de HTTPS ativado: ícone de cadeado.

- Status do GPS:

- Hora e Minuto do terminal: **HH:MM**

- Status da COMUNICAÇÃO: **XXXX**

O status da comunicação será apresentado em quatro caracteres com as seguintes informações:

```

"Init " = estado inicial antes de ativar o TCPIP ao iniciar o POS
"Strt " = estado após iniciar o TCPIP - aguardando a rede entrar no ar
"Open " = dispositivo aberto e prestes a entrar no ar
"Up    " = rede no ar e operacional
"Down " = rede caiu - recuperando-se
"Clse " = dispositivo fechado - o NAVS vai reiniciá-lo (ocorre em falha
de rede)
"Out   " = cabo Ethernet desconectado
"Ip    " = resolvendo o IP
  
```

## Especificação de Interface

---

"Cnct " = conectando ao socket  
"Send " = enviando dados  
"Recv " = recebendo dados  
"Host " = aguardando o host enviar os dados

A partir da versão 1.19.1, se o POS tem 2 tipos de comunicação: WiFi e GPRS, ou GPRS e ETH, é possível habilitar automaticamente os dois tipos.

Nova variável foi criada para este controle: METHOD2.

Setando os 2 tipos, o status da comunicação indicará os dois dispositivos em 2 dígitos.

Expo:

"InCn" : primeira conexão está em "Init" e a segunda em "Cnct".

Versões anteriores à 1.12.0:

**CX..:** Conectando na rede.  
**Rdok:** Rede OK, indica conexão física com sucesso.  
**StNw:** Inicia conexão lógica.  
**Cxok:** Conexão lógica com sucesso, conectado com a operadora.  
**Wfok:** Encontrou a rede Wi-Fi.  
**Htok:** Conectado com o Servidor.  
**HOST:** Servidor OK.  
**Hoff:** Servidor offline.  
**RX..:** Recebendo dados.  
**Rxok:** Dados do Servidor recebidos.  
**TX..:** Enviando dados ao Servidor.  
**CXer:** Erro de conexão.  
**DNSe:** Erro de DNS, tentando encontrar página.  
**....:** Reconectando na rede.

## 16.0 Histórico das Versões

NAVS	Alteração	Autor/Revisor
1.4.8 03/06/2013	<ul style="list-style-type: none"> <li>- Touch no Vx680: comandos para desenho de figuras e captura.</li> <li>- Ajuste na leitura do cartão magnético.</li> <li>- DNS na configuração do acesso ao Servidor.</li> <li>- Ajuste no HTTPS.</li> <li>- Paginação de tela.</li> </ul>	Orlando Palermo
1.4.9 14/06/2013	<ul style="list-style-type: none"> <li>- Ajuste no Touch com CAPTURE.</li> <li>- Ajuste no HTTPS para o comando FILE.</li> <li>- Comando para captura de data e hora (GET DATETIME).</li> <li>- Ajuste no comando SERIAL com timeout configurado (SER-TIMEOUT).</li> <li>- Aumento no tamanho de entrada dos dados na Configuração.</li> <li>- Ping para manter conexão ativa no GPRS, configurável.</li> </ul>	Orlando Palermo
1.5.2 31/07/2013	<ul style="list-style-type: none"> <li>- Aumento do limite de quantidade de imagens para a tela para 50.</li> <li>- Comando SERIAL sem controle de Timeout.</li> <li>- Comando DELFILE para remover arquivos no POS.</li> <li>- Comando de impressão de Boleto.</li> <li>- Comandos para OFFLINE, capítulo 4.0.</li> </ul>	Orlando Palermo
1.5.4 30/08/2013	<ul style="list-style-type: none"> <li>- Comando de impressão de Boleto, com status de retorno.</li> <li>- Suporte a cookies na comunicação HTTP.</li> </ul>	Orlando Palermo
1.5.5 06/09/2013	<ul style="list-style-type: none"> <li>- Comando CHGPRNFNT: novo comando com diversas fontes para impressora.</li> <li>- Comando SELECT, para escolha de item em menus.</li> </ul>	Orlando Palermo

## Especificação de Interface

NAVS	Alteração	Autor/Revisor
1.5.7 16/09/2013	<ul style="list-style-type: none"> <li>- Comando FILE com novo parâmetro WRT, para opção de sobrescrever ou não um arquivo já existente.</li> <li>- Apresentação da Hora e Minuto na linha de status</li> </ul>	Orlando Palermo
1.5.9 26/09/2013	<ul style="list-style-type: none"> <li>- Comando GET FIELD com ALPHA=1, aceitando espaços.</li> <li>- Adição de um parâmetro de configuração para o método HTTPS: "SSL_METHOD" que recebe valores 1 ou 2, onde 1 (indica que o protocolo utilizado será o SSLv3, mais recomendável por sua confiabilidade) e 2 (indica que o protocolo utilizado será TLSv1).</li> </ul>	Thais Marques
1.6.1 01/11/2013	<ul style="list-style-type: none"> <li>- Comando para captura da versão do Sistema Operacional do POS (GET OSVERSION).</li> <li>- Comando para captura de CPF, CNPJ e Data.</li> <li>- Aumento do limite dos itens de menu do comando GET SELECT, com a criação de um novo comando: "SELECT".</li> <li>- Comando TONE: novo comando que faz o POS emitir sons;</li> <li>- Adição do parâmetro de configuração do nível de contraste na impressora no menu de configurações do NAVS.</li> </ul>	Thais Marques
1.6.2 12/11/2013	<ul style="list-style-type: none"> <li>- Correção no retorno do comando SELECT.</li> <li>- Correção na estrutura de navegação do NAVS, para que as telas não travem quando o botão cancela for pressionado enquanto o comando SELECT está ativo.</li> <li>- Correção do comando CAPTURE.</li> </ul>	Thais Marques
1.6.3 13/12/2013	<ul style="list-style-type: none"> <li>- Adição de status de retorno no comando &lt;PRINTER&gt;;</li> <li>- Adições no menu de configuração do NAVS: (Opção para ativar pagamento);</li> <li>- Opção de mostrar ou não hora na linha de status; Adição do caractere "_" no teclado, pode ser inserido quando a tecla "0" é apertada e logo em seguida a tecla "#" é apertada 3 vezes.;</li> <li>- Adição do caractere "@" no teclado, pode ser inserido quando a tecla "0" é apertada e logo em seguida a tecla "#" é apertada 4 vezes;</li> <li>- Adição do comando SET TIME;</li> </ul>	Thais Marques

NAVS	Alteração	Autor/Revisor
	<ul style="list-style-type: none"> <li>- Alteração no comando SENDFILE para enviar arquivos não texto, e removida o parâmetro de retorno.</li> <li>- Ajuste nos comandos para captura de CPF, CNPJ e Data.               <ul style="list-style-type: none"> <li>- Offline com LUA / Sqlite.</li> </ul> </li> </ul>	
1.6.4 18/12/2013	<ul style="list-style-type: none"> <li>- Novo ajuste nos comandos para captura de CPF, CNPJ e Data.</li> <li>- Observações para utilização dos comandos SELECT, CONSOLE, PRINTER, PRNFNT e CHGPRNFNT.</li> </ul>	Thais Marques
1.6.5 26/12/2013	<ul style="list-style-type: none"> <li>- Nova correção nos comandos para captura de CPF, CNPJ e Data.</li> <li>- Correção para o uso do comando SELECT junto com o comando INIT.</li> <li>- Correção no comando SELECT para aceitar as teclas numéricas como teclas de navegação, antes essas teclas não eram capturadas.</li> </ul>	Thais Marques
1.6.6 09/01/2014	<ul style="list-style-type: none"> <li>- Ajuste no comando GET FIELD com ALPHA=1, quando usa-se o caracter de espaço.</li> <li>- Especificação atualizada com a informação do uso do parâmetro PING=1 na conexão Wi-Fi.</li> </ul>	Orlando Palermo
1.6.7 16/01/2014	<ul style="list-style-type: none"> <li>- Adição do parâmetro HOST na configuração do servidor.</li> <li>- Adição do caractere "!" no teclado, pode ser inserido quando a tecla "0" é apertada e logo em seguida a tecla "#" é apertada 5 vezes;</li> <li>- Adição de um parâmetro de entrada NOCLS para o comando CONSOLE, para não limpar a tela.</li> </ul>	Thais Marques
1.6.9 12/02/2014	<ul style="list-style-type: none"> <li>- Correção do comportamento do &lt;CONSOLE&gt; ao tratar o parâmetro NOCLS.</li> <li>- Correções no comando &lt;SET TYPE=TIME ..&gt;, captura de dados, e travamento dos POS nas operações posteriores no POS.</li> <li>- Correções no comando &lt;SELECT&gt; (aceita "*" e "#" como parâmetros, nova correção de uso junto ao comando &lt;INIT&gt;, e atualização da documentação referente ao mesmo).</li> </ul>	Thais Marques



NAVS	Alteração	Autor/Revisor
	<ul style="list-style-type: none"> <li>- Correção do comando de Scroll (uso exclusivo do Vx520).</li> <li>- Adição de uma mensagem de status de conexão.</li> <li>- Identificação do tipo de aparelho e disponibilização da configuração de rede conforme as possibilidades do dispositivo.</li> </ul>	
1.7.1 10/03/2014	<ul style="list-style-type: none"> <li>- Inclusão no manual NAVS os capítulos 3.2.19 e 5.0, com comandos de leitura e escrita no Mifare.</li> <li>- Criação do comando KEYBOARD.</li> <li>- Suporte NAVS para aparelho Vx520 Contactless da Verifone.</li> <li>- Determinação de padrões de configuração automática da operadora, quando a conexão for GPRS.</li> <li>- Informações adicionais sobre imagens envolvendo os comandos CONLOGO, PRNLOGO.</li> </ul>	Orlando Palermo e Thais Marques
1.7.2 17/03/2014	<ul style="list-style-type: none"> <li>- Correção das informações sobre o comando PRNLOGO (especificação de resolução máxima da imagem de 128x64 para 384x240).</li> <li>- Adição de informações no comando CONLOGO.</li> <li>- Melhoria na velocidade de impressão.</li> </ul>	Thais Marques
1.7.4 30/04/2014	<ul style="list-style-type: none"> <li>- Adição do comando BINS CARD.</li> <li>- Adição do comando APDUCHIP.</li> <li>- Adição do comando DESNBS.</li> <li>- Adição do comando WRITE_AT.</li> <li>- Adição de novos fontes para o console e como consequência a criação do comando CHGCONFNT, que serve para alternar entre os fontes.</li> <li>- Adição do comando CONFIG_NAVS.</li> <li>- Correção no controle de funcionamento da impressora que permite que ela continue imprimindo após o comandos relacionados á impressão.</li> <li>- Adição de parâmetro no comando PRINTER (ENABLE_CANCEL).</li> </ul>	Thais Marques

NAVS	Alteração	Autor/Revisor
	<ul style="list-style-type: none"> <li>- Adição de parâmetro no comando INIT (KEEP_COOKIES).</li> <li>- Melhoria de performance do Lua.</li> <li>- Implementação de Log para auxiliar na programação Lua.</li> <li>- Ajuste de sincronia entre LUA e o NAVS, quando o botão CANCEL é acionado.</li> </ul>	
1.7.6 28/05/2014	<ul style="list-style-type: none"> <li>- Correção no comando SELECT.</li> <li>- Correção do comando de debug do LUA.</li> <li>- Adição do comando HTMLDEBUG.</li> <li>- Adição do comando de captura de horário com máscara (GET TYPE=HOUR).</li> <li>- Correção na recepção de parâmetros do KEEP_COOKIES (no comando INIT).</li> <li>- Inclusão dos parâmetros APDU1, APDU2, APDU3, APDU4 e APDU5 no comando APDUCHIP com intuito de ampliar o tamanho da APDU a ser executada.</li> <li>-Ajuste no comando dofile.</li> </ul>	Thais Marques
1.7.7 06/06/2014	<ul style="list-style-type: none"> <li>- Correção no comando SENDFILE para trabalhar com arquivos maiores que 32kbytes.</li> <li>- Nova alteração do comando de debug do lua (alteração de sintaxe também).</li> <li>- Atualização na parte de comunicação para que o Wi-fi também funcione com SSID oculto (somente para os modelos Verifone vx680 Wi-fi).</li> <li>- Disponibilização da ferramenta de encriptação de scripts feitos em lua.</li> <li>- Atualização da biblioteca de boleto.</li> </ul>	Thais Marques
1.7.8 24/07/2014	<ul style="list-style-type: none"> <li>- Adição do comando que cria QR Codes a partir de strings.</li> <li>- Adição do comando que gera código de barras 2 de 5 (ITF).</li> <li>- Adição do comando que gera código de barras 3 de 9.</li> <li>- Correção no comando APDUCHIP.</li> </ul>	Thais Marques

## Especificação de Interface

NAVS	Alteração	Autor/Revisor
	<ul style="list-style-type: none"> <li>- Adição do Keep Alive nas configurações NAVS.</li> <li>- Nova correção no SENDFILE para evitar que os arquivos sejam corrompidos.</li> <li>- Correção no GET TYPE=CPF e GET TYPE=CNPJ.</li> </ul>	
1.7.9 08/08/2014	<ul style="list-style-type: none"> <li>- Adição do comando DRAW.</li> <li>- Correção e adição de 2 novas funcionalidades ao SELECT.</li> <li>- Alterações no modo de funcionamento do comando CONFIG_NAVS.</li> <li>- Mudança de nome de alguns parâmetros de configuração do KEEP ALIVE.</li> </ul>	Thais Marques
1.8.0 12/08/2014	<ul style="list-style-type: none"> <li>- Ajustes na interface de comunicação, ficando funcional a comunicação ETH usando IP fixo, com DNS externo.</li> </ul>	Orlando Palermo
1.8.1 15/09/2014	<ul style="list-style-type: none"> <li>- Correções no funcionamento entre o CAPTURE e o TIMEOUT, eliminando o erro de memória.</li> <li>- Correções no SELECT para eliminar também um erro de memória, relacionado ao comando INIT.</li> </ul>	Thais Marques
1.8.3 14/10/2014	<ul style="list-style-type: none"> <li>- Melhoria de 155performance.</li> <li>- Implementação do Menu que informa as versões de software que estão no POS.</li> <li>- Novas correções no comando SELECT.</li> </ul>	Thais Marques
1.8.5 14/11/2014	<ul style="list-style-type: none"> <li>- Aumento da limitação do campo SERVER_HOST do comando Config_Navs ;</li> </ul>	Thais Marques
1.8.6 20/11/2014	<ul style="list-style-type: none"> <li>- Nova Correção do SELECT relacionado ao uso conjunto com o comando INIT.</li> </ul>	Thais Marques
1.8.7 19/12/2014	<ul style="list-style-type: none"> <li>- Ajuste na comunicação GPRS para manter por mais tempo a conexão com a operadora, mesmo que o PING estiver definido com um valor alto.</li> </ul>	Thais Marques

NAVS	Alteração	Autor/Revisor
	<ul style="list-style-type: none"> <li>- Ajuste na comunicação de rede wireless para funcionar em rede sem senha.</li> <li>- Adição de novos caracteres para a configuração NAVS. Os caracteres adicionados foram : '#','\$','(',')','&amp;','+','?',' ',';',':','"','\',''','{','}','[','\'],'=','"',',','%'.</li> <li>- Ajuste na exibição do texto capturado pelo comando KEYBOARD.</li> <li>- Correção do comportamento do comando SELECT com a alteração do posicionamento da linha de status.</li> <li>- Adição do comando que retorna o status atual de comunicação (GET TYPE=STSCOMM).</li> <li>- Ajuste na especificação do comando "GET TYPE=ANYKEY".</li> <li>- Adição de conteúdo na especificação do comando Console para inserção de caracteres especiais na tela.</li> </ul>	
1.8.8 09/01/2015	<ul style="list-style-type: none"> <li>- Mudança na estrutura de resposta do comando Config_NAVS.</li> <li>- Melhorias para exibição na tela, reduzindo o número de vezes que a tela pisca.</li> <li>- Ajuste na gravação de parâmetros do menu de configuração NAVS.</li> </ul>	Thais Marques
1.8.9 06/02/2015	<ul style="list-style-type: none"> <li>- Ajuste no status de fim de papel no Lua.</li> <li>- Adição de informações sobre a utilização do Lua.</li> </ul>	Orlando Palermo
1.9.0 12/02/2015	<ul style="list-style-type: none"> <li>- Criação do comando de download e upload via FTP.</li> <li>- Criação do comando de leitura biométrica (FINGERPRINT).</li> <li>- Correção no comando Config_NAVS(bug na gravação do parâmetro de recurso do servidor).</li> </ul>	Thais Marques
1.9.1 25/03/2015	<ul style="list-style-type: none"> <li>- Criação da Mascara Genérica, acionada através do comando GET TYPE=FIELD com o parâmetro MASK.</li> <li>- Comando Config_NAVS possui a opção de saber quais os dados configurados no POS com a adição de um GET no cabeçalho da requisição.</li> <li>- A configuração de parâmetros da SERIAL agora também pode ser feita remotamente através do comando Config_NAVS.</li> </ul>	Thais Marques

## Especificação de Interface

NAVS	Alteração	Autor/Revisor
	<ul style="list-style-type: none"> <li>- Correção para o funcionamento das teclas de scroll no VX520(Verifone).</li> <li>- Correção de alguns pontos de memory leak.</li> <li>- Adição de informações de uso do comando FILE em Java.</li> </ul>	
1.9.2 22/05/2015	<ul style="list-style-type: none"> <li>- Ajuste do comando SELECT e do CONSOLE para que preencham a linha de status quando ela estiver ausente.</li> <li>- Correção no comando SELECT para listar mais 41 itens.</li> <li>- A partir dessa versão o NAVS passa a ser suportado no modelo Vx685 da Verifone.</li> <li>- Comando GET EMV, para leitura de cartões com chip EMV.</li> <li>- O NAVS agora funciona com certificado particular( local ) para SSL.</li> </ul>	Thais Marques
1.9.3 18/06/2015	<ul style="list-style-type: none"> <li>- Comando que gera o código de barras 2 de 5 (ITF) agora possui o parâmetro SPACES para ajuste na impressão.</li> <li>- Comando Keyboard agora tem um parâmetro que permite o alinhamento de texto (direita, esquerda, centro).</li> <li>- Criação do comando GET TYPE=INFO_FILE.</li> <li>- Adição do parâmetro ZL para o "Get"s que aceitam input numérico do teclado, para que ignorem os zeros à esquerda.</li> <li>- Correção na parte gráfica.</li> <li>- Correções de bugs do Lua.</li> </ul>	Thais Marques
1.9.4 24/08/2015	<ul style="list-style-type: none"> <li>- Os comandos GET TYPE=IDMIFRAE, CTLSWRITE e GET TYPE=MIFARE_READ agora possuem o parâmetro RC_NAME=vvv o qual indica o nome de uma variável para receber o retorno do comando (que deve ser '00' se o comando for ok).</li> </ul>	José Rivera
1.9.5 11/09/2015	<ul style="list-style-type: none"> <li>- Correção de falha na conexão do LUA com rede</li> </ul>	José Rivera

## Especificação de Interface

NAVS	Alteração	Autor/Revisor
1.9.6 29/09/2015	<ul style="list-style-type: none"> <li>- Abertura do dispositivo de cartão tarja, após uso do comando de Pagamento.</li> <li>- Alteração na ordem dos recibos de pagamento.</li> <li>- Fonte de impressão de pagamento com opção Ampliada.</li> </ul>	Orlando Palermo
1.9.7 30/10/2015	<ul style="list-style-type: none"> <li>- Adição do parâmetro TIMEOUT ao comando POST, assim se o servidor não responder depois de um determinado tempo executa as ordens que virão depois do comando.</li> <li>- Validação de licença do POS Online, assim não é necessário mais digitar a senha para ativar o NAVS.</li> <li>- Correção no comando CONFIG_NAVS para alterar o parâmetro indicado em POSITION_STATUS_LINE.</li> <li>- Melhoria da especificação para os comandos relacionados á leitura biométrica.</li> </ul>	Thais Marques
1.9.8 07/12/2015	<ul style="list-style-type: none"> <li>- Ajuste no Aviso da licença online</li> </ul>	Orlando Palermo
1.9.9 23/12/2015	<ul style="list-style-type: none"> <li>- Uso da variável de ambiente DEBUS = 1, para uso do Trace via USB.</li> <li>- Correção para habilitação da leitura Tarja após uso do comando GET TYPE=EMV.</li> <li>- Comando PAGAMENTO com TIPO genérico (numérico).</li> <li>- Inclusão da opção CANAL nas configurações da rede wi-fi.</li> <li>- Criado o comando UNZIP.</li> </ul>	Orlando Palermo
1.10.0 11/01/2016	<ul style="list-style-type: none"> <li>- Incluído código de erro 13 neste documento no comando EMV.</li> <li>- Comandos para o Lua: and, or, xor.</li> <li>- Novos comandos de debug do lua (cap. 4.0)</li> </ul>	Orlando Palermo
1.10.1 01/02/2016	<ul style="list-style-type: none"> <li>- Aumento na quantidade de imagens na console, de 50 para 300.</li> </ul>	Orlando Palermo
1.10.2 24/02/2016	<ul style="list-style-type: none"> <li>- Criado configuração no NAVS para não imprimir recibo do Pagamento.</li> <li>- Porting para os modelos Vx690 e E265 da Verifone.</li> </ul>	Orlando Palermo
1.10.3 08/03/2016	<ul style="list-style-type: none"> <li>-Correção de erro LUA para o comando _debug_off()</li> </ul>	José Rivera
1.10.4 18/03/2016	<ul style="list-style-type: none"> <li>- Inclusão do comando PAGAMENTO neste documento</li> <li>-Inclusão dos comandos "SET TYPE=BACKLIGHT" e "SET TYPE=KBBACKLIGHT"</li> <li>-Inclusão da intensidade da luz de fundo da tela e do teclado na</li> </ul>	José Rivera

## Especificação de Interface

NAVS	Alteração	Autor/Revisor
	configuração -Possibilidade de mostrar o status da bateria em percentual -Configuração de nível de carga de bateria para exibição de ícone de alarme -Melhorias no consumo de bateria pela função de CTLS	
1.10.5 24/03/2016	-Possibilidade de o terminal entrar em hibernação para economia de energia -Desligamento automático após tempo em hibernação -Restrição de 3 níveis de contraste de impressão para os terminais VX685 e VX690 -Apagamento de tabelas Clisitef -Novos códigos de erro Clisitef	José Rivera
1.10.6 30/03/2016	- Ajuste retorno função configuração Sitef -Inclusão da identificação de chip Vodafone na linha de status -Ajuste na linha de status do E265	Orlando Palermo
1.10.7 04/04/2016	- Nova Clisitef e V-Lib para comando PAGAMENTO	Orlando Palermo
1.10.8 18/04/2016	- Alterações das mensagens no LOL - Novo comando ID CTLS para cartões sem contato ISO - Ajuste na configuração da comunicação para o Vx520G (novo modem) - Inclusão de novos parâmetros no CONFIG_NAVS - Ajuste no Debug Lua - Inclusão dos retornos de Rede Adquirente e Tipo de Cartão no Pagamento	Orlando Palermo
1.10.9 03/05/2016	- Ajustes no processo de checagem da licença on-line -Novas mensagens da licença on-line -Informações do licenciamento online disponíveis no terminal -Possibilidade de desativar o comando pagamento -Guarda em arquivo dos recibos de pagamento -Incluído neste documento os Erros do protocolo HTTP -Data do POS sincronizada automaticamente com Servidor LOL, configurável na opção Licenciamento POS	José Rivera
1.11.0 02/06/2016	- Ajuste nos arquivos dos recibos de Pagamento: sem nulos no final. - DNS no IPTEF do comando PAGAMENTO. - Comando KEYBOARD com caracteres especiais. - LOL automático, sem ação do usuário (configurável). - Modelo do POS no LOL. - Nova LIB EOS SDK. - Ajuste no comando sock.recv com terminador. - Beep no comando FINGERPRINT.	Orlando Palermo
1.11.1 24/06/2016	- Ajuste no Pagamento nos códigos de retorno da comunicação. - Ajuste FTP Passivo. - Nova Lib Boleto, aumento campo NOSSO NUM	Orlando Palermo

## Especificação de Interface

NAVS	Alteração	Autor/Revisor
1.11.2 08/07/2016	<ul style="list-style-type: none"> <li>- Ajuste status bateria no Vx685.</li> <li>- DNS para uso na conexão via LUA.</li> <li>- Ajuste comando Get EMV.</li> <li>- Ajuste no uso MIFARE com PAGAMENTO.</li> <li>- Identificação de erro na conexão antes de iniciar o PAGAMENTO.</li> <li>- Ajuste FTP Passivo.</li> </ul>	Orlando Palermo
1.11.3 27/07/2016	<ul style="list-style-type: none"> <li>- Correção senha online para Pagamento: nova V-Lib Verifone.</li> <li>- Validação de Valor zerado no Pagamento.</li> <li>- Ajuste FTP Passivo PUT.</li> </ul>	Orlando Palermo
1.11.4 05/08/2016	<ul style="list-style-type: none"> <li>- NOENTER = 1 incluído nos comandos: CPF, VALUE, CNPJ, DATA, HOUR e PASS.</li> <li>- Comando CAPTURE com retorno da identificação do GET que foi ativado.</li> <li>- Aumento dos buffers de leitura e escrita dos comandos MIFARE.</li> </ul>	Orlando Palermo
1.11.5 e 1.20.2(SSL) 09/08/2016	<ul style="list-style-type: none"> <li>- Ajuste comando PAGAMENTO na impressão e reimpressão dos comprovantes.</li> </ul>	Orlando Palermo
1.12.0 17/08/2016	<ul style="list-style-type: none"> <li>- Nova interface de comunicação.</li> <li>- Licença Pagamento Online.</li> <li>- Ajustes para FTP Passivo, Keep Alive e Economia de energia.</li> </ul>	Orlando Palermo
1.12.1 26/08/2016	<ul style="list-style-type: none"> <li>- Ajuste ETH com DHCP, cookies Java, alocação memória Pagamento.</li> <li>- Novos parâmetros do GET SPECIALKEY: NOENTER e NOBKSPC.</li> </ul>	Orlando Palermo
1.12.2 02/09/2016	<ul style="list-style-type: none"> <li>- Ajuste POST com TIMEOUT.</li> <li>- Config NAVS com parâmetros do LOL.</li> <li>- Melhoria no desempenho da comunicação WiFi.</li> <li>- Ajuste para mostrar corretamente os erros do HTTP.</li> </ul>	Orlando Palermo
1.12.3 09/09/2016	<ul style="list-style-type: none"> <li>- Mudanças na Configuração do NAVS.</li> <li>- Pagamento sem parâmetros do CB.</li> </ul>	Orlando Palermo
1.15.0 28/09/2016	<ul style="list-style-type: none"> <li>- Clisitef Integrada, com e sem SSL e com CB.</li> <li>- Ajuste comando GET ICCID.</li> <li>- Mudanças na configuração do NAVS, Rede e Pagamento.</li> <li>- Ajustes comunicação evitando reinício do POS.</li> <li>- Comando KEYBOARD com digitação a partir da segunda linha.</li> <li>- Ponto da impressão do Pagamento definido por configuração NAVS.</li> <li>- Incluído padrões Lyra e Vodafone na configuração GPRS.</li> </ul>	Orlando Palermo
1.16.0 26/10/2016	<ul style="list-style-type: none"> <li>- Comando para captura do % da bateria (GET TYPE=BATTERY_STATUS).</li> <li>- Comando para impressão de código de barras – CODE-128 tipos A, B e C.</li> <li>- Correção de DNS Server para alguns chips GPRS M2M.</li> <li>- Captura do código OTP até 30 dígitos.</li> </ul>	José Rivera



## Especificação de Interface

NAVS	Alteração	Autor/Revisor
1.16.1 04/11/2016	<ul style="list-style-type: none"> <li>- Parâmetros de saída do comando pagamento agora opcionais</li> <li>- Inclusão dos parâmetros VCANCEL, NOPRINT, RECLOJA e RECCLI no comando pagamento</li> </ul>	José Luiz
1.16.3 17/11/2016	<ul style="list-style-type: none"> <li>- Atualização da Clisitef integrada, corrigido timeout quando Sitef não responde.</li> <li>- Teste de chaves também no Display do POS.</li> <li>- Mudança na mensagem na mudança do status do Pagamento.</li> <li>- Senha supervisor na Reimpressão do Pagamento como opcional.</li> <li>- Ajuste na rotina de Receive da comunicação para corrigir eventuais reinicializações do POS ao receber dados.</li> <li>- Alteração nos parâmetros do comando QR CODE.</li> </ul>	Orlando Palermo
1.16.4 24/11/2016	<ul style="list-style-type: none"> <li>- Clisitef com correção no SSL, entrada do OTP.</li> <li>- Inclusão do parâmetro de Ativação do SSL no comando CONFIG_NAVS.</li> <li>- Ajuste nas confirmações da função Finaliza Pagamento no tratamento dos recibos.</li> <li>- Ajuste ao configurar ETH modo estático.</li> </ul>	Orlando Palermo
1.16.5 28/11/2016	<ul style="list-style-type: none"> <li>- Redes Wi-Fi inteligente, mostra a lista das redes para escolha.</li> </ul>	Orlando Palermo
1.16.6 30/11/2016	<ul style="list-style-type: none"> <li>- Ajuste em exceções no processo WiFi inteligente.</li> <li>- Alteração na mensagem de chaves ausentes no Teste de Chaves.</li> </ul>	Orlando Palermo
1.16.7 06/12/2016	<ul style="list-style-type: none"> <li>- Ajuste comandos CPF, CNPJ, DATA e HORA após apagar dígitos.</li> <li>- Ajuste HOST vazio – erro 400.</li> <li>- Inversão na lógica da opção Senha Supervisor na Reimpressão.</li> </ul>	Orlando Palermo
1.16.8 27/12/2016	<ul style="list-style-type: none"> <li>- Confirmação de Transação Pendente no Pagamento, habilitado pela configuração do NAVS.</li> <li>- Compatibilidade com os modelos Verifone: Vx820 e C680.</li> <li>- Novo comando DEVICE, para gravação, leitura e remoção de arquivos em dispositivos externos, como Pendrive e SD.</li> <li>- Apn, User e Psw no GPRS como case sensitive.</li> <li>- Arquivos dos recibos de Pagamento sempre salvos, independente da opção Não Imprime dos Recibos.</li> <li>- Novo comando GET TYPE=DRIVE_SIZE, retorna memória total e em uso do drive solicitado.</li> <li>- Ajuste no comandos Get, evitando reinício.</li> <li>- Ajuste mensagem de erro Lua na Console.</li> </ul>	Orlando Palermo
1.16.9 05/01/2017	<ul style="list-style-type: none"> <li>- Teste de Chaves mais completo.</li> <li>- Ajuste na passagem de variáveis no Lua.</li> <li>- Melhoria interna do comando UNZIP.</li> <li>- Tratamento para Dual Chip, na configuração e através de comando: GPRS_SWITCH_SIM</li> <li>- Comando KEYBOARD com novo modelo A1.</li> </ul>	Orlando Palermo
1.17.0 12/01/2017	<ul style="list-style-type: none"> <li>- Rede Wi-Fi invisível para WPA e WPA2.</li> <li>- Ajuste comando CODE128 no Lua.</li> <li>- Ajuste HTTPS e inclusão do parâmetro SNI.</li> <li>- Comando WRITE_AT e comandos GETs: ocupa apenas os espaços do texto escrito (não apaga mais a linha toda).</li> <li>- Ajuste no display do teste de chaves.</li> </ul>	Orlando Palermo

## Especificação de Interface

NAVS	Alteração	Autor/Revisor
1.17.1 20/01/2017	<ul style="list-style-type: none"> <li>- Novo comando CANCEL_KEY, para inibir a tecla Cancel.</li> <li>- Alteração nas mensagens de comunicação.</li> <li>- Ajuste https no LOL e Pagamento.</li> </ul>	Orlando Palermo
1.17.2 26/01/2017	<ul style="list-style-type: none"> <li>- Ajustes diversos: erros http, https comando FILE (reinício).</li> <li>- Comando SELECT com até 70 caracteres na linha.</li> <li>- Configuração para Dual Chip apenas nos modelos que suportam.</li> </ul>	Orlando Palermo
1.17.3 03/02/2017	<ul style="list-style-type: none"> <li>- Ajuste na alocação de memória para o LUA e criação de uma nova variável de ambiente: LUA_LOW_STACK (1 para diminuir a memória e 0 para manter o tamanho de memória atual)</li> <li>- Ajuste no limite do tamanho do QRCODE (aumentado de 23 bytes para 200 bytes),</li> <li>- Ajuste no manual: mensagem de retornos HTTP</li> </ul>	Fabio Ishikawa
1.17.4 27/02/2017	<ul style="list-style-type: none"> <li>- Ajuste conflito Mifare.</li> <li>- Ajuste em falha de impressão via Loja no PAGAMENTO.</li> </ul>	Orlando Palermo
1.18.0 07/03/2017	<ul style="list-style-type: none"> <li>- OTP automático quando SSL ativo.</li> <li>- Após o licenciamento mostra nome cliente (e imprime) e status do pagamento e otp.</li> <li>- Mensagens do LOL desabilitadas por configuração.</li> <li>- Libs para Lua integradas na carga do NAVS: comunica.o e navs.o (veja detalhe no cap.4.0).</li> <li>- Novo comando para relicenciar o POS: &lt;RELICENSE&gt;.</li> <li>- Ajustes em comandos de entrada e máscaras.</li> </ul>	Orlando Palermo
1.18.1 14/03/2017	<ul style="list-style-type: none"> <li>- Comando para mostrar GIF's na tela (ANIMATE).</li> <li>- Padrão da carga, sem mensagens do LOL.</li> <li>- Comando POST com retorno dos erros e parâmetros de conexão: ip, porta, host e recurso.</li> <li>- Comando PAGAMENTO com entrada para número de parcelas, se venda parcelada.</li> <li>- Confirmação do Pagamento realizado ao receber os dados do recibo.</li> <li>- Ajuste busca de MK para envio ao LOL.</li> <li>- Mensagem 3G na linha de status quando chip conectar em 3G.</li> <li>- Ícone da bateria na linha de status indica a carga quando o POS estiver carregando.</li> </ul>	Orlando Palermo
1.18.2 20/03/2017	<ul style="list-style-type: none"> <li>- Modo 2G, 3G por configuração.</li> <li>- Pagamento com transação direta para Venda Parcelada.</li> <li>- Ajuste Mifare no CANCEL.</li> <li>- Ajuste no envio do KSI no LOL.</li> <li>- Ajuste no LOL com chave antiga da Global.</li> </ul>	Orlando Palermo
1.18.3 28/03/2017	<ul style="list-style-type: none"> <li>- Reconexão automática Wi-Fi em queda de sinal (Reconexao ponto de acesso).</li> <li>- Ajuste Mifare.</li> <li>- Ajuste parâmetro NOPRINT no comando PAGAMENTO.</li> </ul>	Orlando Palermo
1.18.4 18/04/2017	<ul style="list-style-type: none"> <li>- PAGAMENTO com entrada direta para reimpressão específica, OUTROS.</li> <li>- Lua com 2 novos comandos: URL_encode e URL_decode.</li> <li>- Ajuste comando de captura com máscara genérica.</li> <li>- Ajuste Timeout do comando Capture.</li> </ul>	Orlando Palermo

## Especificação de Interface

NAVS	Alteração	Autor/Revisor
1.18.5 25/04/2017	<ul style="list-style-type: none"> <li>- Pagamento retornando código de erro, quando transação não aprovada.</li> <li>- Ajuste no comando WRITE_AT no Vx690.</li> <li>- Ajuste na reimpressão específica (Pagamento) quando ponto da impressão é antes de retirar o cartão.</li> <li>- Comando KEYBOARD, com parâmetro inicial de <u>teclado</u> e máscara.</li> </ul>	Orlando Palermo
1.18.6 28/04/2017	<ul style="list-style-type: none"> <li>- Número de parâmetros nos comandos NAVS ampliado para 20 (era 15).</li> <li>- Quantidade de caracteres nos parâmetros dos comandos NAVS ampliado para 240 (era 150).</li> </ul>	Orlando Palermo
1.18.7 03/05/2017	<ul style="list-style-type: none"> <li>- Ajuste comando CONSOLE quando usado em conjunto com o WRITE_AT.</li> <li>- Novos parâmetros no comando CONFIG_NAVS.</li> <li>- Novos parâmetros de retorno no PAGAMENTO: NSUSITEF e NSUCANC.</li> <li>- Ajuste tecla BackSpace no comando MASK.</li> </ul>	Orlando Palermo
1.19.1 12/05/2017	<ul style="list-style-type: none"> <li>- Modo automático WiFi e GPRS, ETH e GPRS.</li> <li>- Otimização no uso das máquinas de estado do NAVS.</li> <li>- Ajuste comando SERIAL para impressora externa.</li> <li>- Ajuste comando MASK: quando tinha outro Get Field antes.</li> <li>- Comando Pagamento, com Débito Parcelado acionado pelo parâmetro NPAR.</li> <li>- Se NPAR=1 no Pagamento, será definido como à Vista.</li> <li>- Parâmetros novos GPRS_SIMS, METHOD2, DISABLE_GPRS no CONFIG_NAVS.</li> <li>- POS sem chip não tenta a reconexão (não reconecta automático se inserir chip com POS ligado), com variável DISABLE_GPRS = S ou 1.</li> <li>- Se SSL ativado, e erro no OTP, não permite o Licenciamento do NAVS.</li> <li>- Ajuste Pagamento Corban, para não finalizar quando tipo 312I ou 313I.</li> <li>- Verificação de string http nula.</li> </ul>	Orlando Palermo
1.19.2 17/05/2017	<ul style="list-style-type: none"> <li>- Ajuste comandos GET: Value, CPF, CNPJ, Data, Hour.</li> <li>- Ajuste na leitura dos parâmetros do comando CONFIG_NAVS.</li> <li>- Variável NAVS_COMM_TCPIP_LEN_RECV = 8192 no config da carga padrão, para melhorar a velocidade na recepção de dados.</li> <li>- Ajuste interface com Pagamento.</li> <li>- Ajuste na recepção de dados para evitar reinício do POS.</li> </ul>	Orlando Palermo
1.19.3 25/05/2017	<ul style="list-style-type: none"> <li>- Novas APNs para GPRS: AVATEK, DATELO e VectoMobile.</li> <li>- Ajuste no HTMLDEBUG.</li> <li>- Ajuste comunicação em lentidão com GPRS.</li> </ul>	Orlando Palermo
1.19.4 16/06/2017	<ul style="list-style-type: none"> <li>- Ajustes: Animate; backspace da MASK; PASS; FILE no https; Capture com SpecialKey e Mifare_read e leitura Mifare.</li> <li>- Novos valores para o parâmetro NOPRINT do Pagamento: "C" ou "L" para não impressão dos recibos Cliente e Loja.</li> </ul>	Orlando Palermo
1.30.8 28/06/2017	<ul style="list-style-type: none"> <li>- Versão com nova interface com Pagamento, para atender tabelas Sitef com Contactless habilitado.</li> <li>- Ajuste Senha Supervisor no Pagamento.</li> </ul>	Orlando Palermo

## Especificação de Interface

NAVS	Alteração	Autor/Revisor
	- Novos parâmetros no Pagamento: CNPJ credenciadora, Header, Footer nos recibos, Cupom fiscal, Fantasia, Endereço, Cidade, CNPJ.	
1.31.0 10/07/2017	- Nova Clisitef que contempla o tratamento para a reinstalação do certificado mediante novas situações de erro do certificado. - Ajuste no backspace da máscara genérica. - Parâmetro MODPAG do Pagamento com texto da modalidade. - Parâmetro NINST do Pagamento ajustado. - Novos parâmetros no Pagamento: CREDLOJ, CREDADM. - Ajuste para habilitar impressora após erro específico de operação cancelada na saída do Pagamento.	Orlando Palermo
1.32.0 19/07/2017	- Nova versão lib Pos Core (4.3.1). - Melhoria comunicação, quando usar 2 tipos de comunicação. - Novo parâmetro na configuração do NAVS, para forçar a desconexão com o Sitef, que passa a ser o padrão.	Orlando Palermo
1.32.1 26/07/2017	- Versão da Lib Pos Core: (4.3.0), para que as imagens do CTLS não apareçam em comandos de MIFARE, somente no comando Pagamento. Caso as imagens não sejam carregadas na pasta F15, estas imagens também não aparecem no Pagamento. - Comunicação igual à versão 1.19.4.	Orlando Palermo
1.32.2 27/07/2017	- Ajuste GET Timeout no Capture.	Orlando Palermo
1.33.0 04/08/2017	- Título SENHA da configuração NAVS mudou para DIGITE O CODIGO. - Ajuste no QR CODE. - Na configuração do Wi-Fi, na apresentação das redes, ao selecionar o X, sai da busca das redes. - Ícone de carga da bateria, quando bateria estiver zerada, mostra o raio. - Ajuste no Pagamento, na validação dos valores mínimo e máximo das parcelas. - Comunicação da versão 1.32.0.	Orlando Palermo
1.34.0 11/08/2017	- Ajuste HTMLDEBUG. - QR Code com buffer local. - Part Number ( que contém a versão PCI) do POS enviado ao LOL. - Ajuste de algumas mensagens da linha de status para o modelo C680. - Ajuste no Footer do Pagamento. - Pagamento grava no arquivo PAGAM.RET os dados do retorno da transação, aprovada ou não aprovada. - IDTERM do comando Pagamento, adotado os 8 últimos dígitos do número de série do POS, caso ele não exista. - Comando SELECT após seleção, não apaga janela do Select. - Tempo de iniciar o NAVS diminuído em 4 segundos. - Ajuste Comunicação.	Orlando Palermo
1.34.1 17/08/2017	- Ajuste no envio da mensagem ao LOL. - Arquivo PAGAM.RET incluído parâmetros CUPOMF e IDTERM, se existirem. - Novo parâmetro na Economia de Energia: Ativar com Eventos de Rede, onde o POS sai do modo de economia quando ocorrer qualquer evento de rede.	Orlando Palermo

## Especificação de Interface

NAVS	Alteração	Autor/Revisor
1.34.2 21/08/2017	- Incluído parâmetros CUPOMF e IDTERM se existirem, no arquivo PAGAM.RET quando transação de Pagamento não for Aprovada.	Orlando Palermo
1.34.3 22/08/2017	- Comando POST com parâmetro KCK para manter a tecla CANCEL desabilitada. - Parâmetro IDTERM do Pagamento desabilitado.	Orlando Palermo
1.34.4 28/08/2017	- Ajuste do assincronismo nos dados criptografados do licenciamento NAVS.	Orlando Palermo
1.35.0 28/09/2017	- Comandos para Web Services. - Comando Crypt, para criptografia AES256 e DESNBS. - Parâmetros de entrada para Estorno no Pagamento: Cupom, Data e Valor. - Novos parâmetros de retorno Pagamento: VPAGO (valor da transação) e FISCAL (cupom fiscal+data e hora fiscal). - Ajustes: para o C680, ícone da bateria quando em percentual, backspace na senha do config NAVS, POST KCK com timeout, alinhamento nome das operadoras no config Navs.	Orlando Palermo
1.36.0 04/10/2017	- GPS para os modelos que suportam - Config NAVS alterado para GPS. - Ajuste CB no Pagamento.	Orlando Palermo