



SmsBr - Soluções em torpedos para celulares
HTTP Integração

Histórico de revisão do documento

Data revisão	Autor	Descrição
17/08/2010	Arthur Henrique Garcia Rego	<i>Adição de exemplo de integração utilizando a linguagem ruby on rail</i>

1. Introdução

Sistema SmsBr oferece vários métodos para enviar e receber mensagens SMS. Este documento contém as especificações para os seguintes métodos:

- Enviar mensagens usando HTTP POST
- Enviar mensagens Formato Binário usando HTTP POST
- Receber mensagens usando HTTP POST
- Receber relatório de entrega HTTP POST
- Consulte relatório de entrega HTTP GET

SmsBr sistema também oferece um solução Network Query. Este é um serviço em relação ao Mobile Number Portability (MNP), permitindo a identificação de rede que um número de celular pertence, e se o número de celular não existe. Por favor, consulte o Capítulo HLR para obter os detalhes da especificação.

2. Enviando Mensagens (SMS)

HTTP POST - Envio único

A URL de destino para a instrução dos dados:

<http://smsbr.com.br/enviosms.php>

Independente da ligação de programação o uso desta url é válido, basta enviar as seguinte variáveis como um POST:

Parâmetros especificações

AUTENTICAÇÃO	chaveAPI	Cliente chave de acesso localizado no topo da página do cliente.
	usuarioNome	Cliente username, login utilizado para o sistema SmsBr
MENSAGEM	numeroTel	Numero do celular de destino, formato internacional (558499009900)
	mensTexto	Mensagem de texto máximo 150 caracteres.

Durante este envio nossos servidores lhe retorna o APIMsgID, que pode ser armazenado para posterior uso, exemplo completo de envio usando PHP:

```

<?php
#-----[Recebendo dados via $_POST]-----
$chaveAPI = $_POST["chaveAPI"]; #Chave API do Usuario
$usuarioNome = $_POST["usuarioNome"]; #Nome de usuario Login
$numeroTel = $_POST["numeroTel"]; #Celular de destino Ex. (558499998888)
$mensTexto = $_POST["mensTexto"]; #Mensagem
$assinatura = $_POST["assinatura"]; #Assinatura da mensagem

$url = "http://smsbr.com.br/enviosms.php";
#-----[Envio dos dados]-----
$sessao_curl = curl_init();
curl_setopt($sessao_curl, CURLOPT_URL, $url);
curl_setopt($sessao_curl, CURLOPT_RETURNTRANSFER, 1);
curl_setopt($sessao_curl, CURLOPT_POST, 1);
curl_setopt($sessao_curl, CURLOPT_POSTFIELDS,
"chaveAPI=$chaveAPI&usuarioNome=$usuarioNome&numeroTel=$numeroTel&mensTe
xto=$mensTexto&assinatura=$assinatura");
$resultado = curl_exec($sessao_curl);
curl_close($sessao_curl);
$APIMsgID= substr($resultado, 0, -1); #Recebe o ID da Mensagem
echo "Sucesso ID: ".$APIMsgID;
?>
  
```

HTTP POST – Envio único Binário

Nosso sistema suporta envio de mensagens codificadas no formato Binário Hexadecimal, para isso basta enviar um post para a url:

<http://smsbr.com.br/enviosmsbinario.php>

Parâmetros especificações

AUTENTICAÇÃO	chaveAPI	Cliente chave de acesso localizado no topo da página do cliente.
	usuarioNome	Cliente username, login utilizado para o sistema SmsBr
MENSAGEM	numeroTel	Numero do celular de destino, formato internacional (558499009900)
	mensTexto	Mensagem Binário (Ex. 7465737465)

Informações sobre codificação: http://sms.24cro.com/op_1_4_en.htm

HTTP POST – Envio em massa

O uso desta URL para envio é destinado a grandes volumes, neste tipo de envio as mensagens são armazenadas em uma fila de entrega, que e processa imediatamente.

URL de destino:

<http://smsbr.com.br/restrito/integracaoGrupoSMS.php>

Independente da linguagem de programação o uso desta url é válido, basta enviar as seguinte variáveis como um POST:

Parâmetros especificações

AUTENTICAÇÃO	api	Cliente chave de acesso localizado no topo da página do cliente.
	login	Cliente username, login utilizado para o sistema SmsBr
	parceiroId	Caso tenha sido indicado por outro usuário e que identificar a mensagem por indicação.
MENSAGEM	celular_numero	Numero do celular de destino, formato internacional (558499009900)
	mensagem	Mensagem de texto máximo 150 caracteres.
	id_propio	ID da mensagem, identificador da mensagem, ID's repetidos não serão enviados

Segue um exemplo completo de envio usando PHP:

```
<?php
$celular_numero = $_POST["celular_numero"]; #Celular Ex. (558499998888)
$mensagem = $_POST["mensagem"]; #Mensagem Maximo 160 Caracteres
$api = $_POST["api "]; #Chave API do Usuario
$login = $_POST["login"]; #Nome de usuario Login
$assinatura = $_POST["assinatura"]; #Assinatura da mensagem
#-----[Gerando o ID propio da mensagem]-----
$numero = rand(00000000,99999999); # Gera o ID: propio da Mensagem
$parceiroId = "00221100"; # ID Parceiro ( Campo não obrigatorio )
$Url = "http://smsbr.com.br/restrito/integracaoGrupoSMS.php";
```

```

#-----[Envio dos dados]-----
$sessao_curl = curl_init();
curl_setopt($sessao_curl, CURLOPT_URL, $Url);
curl_setopt($sessao_curl, CURLOPT_RETURNTRANSFER, 1);
curl_setopt($sessao_curl, CURLOPT_POST, 1);
curl_setopt($sessao_curl,CURLOPT_POSTFIELDS,
"celular_numero=$celular_numero&mensagem=$mensagem&api=$api&login=$login&
id_propio=$numero&parceiroId=$parceiroId&assinatura=$assinatura");
$resultado = curl_exec($sessao_curl);
curl_close($sessao_curl);
echo "ID Mensagem :". $numero;
?>
  
```

Os exemplos acima fazem o envio das mensagens no formato alpha numéricos, o formato do numero do destinatário deve ser formato internacional (558499009900), 55-Pais, 84-DDD, 99009900-Celular de destino.

INTEGRAÇÃO VIA RUBY ON RAIL – Envio único

Para enviar mensagens via Ruby on Rails, deve-se utilizar dos seguintes trechos de código em seu script, geralmente em alguma ação de seu controlador:

```

#requires necessarios, geralmente adicionado no inicio do arquivo
require "uri"
require "net/http"

#recebendo os dados via post
parametros = {'chaveAPI' => params[:chave], 'usuarioNome' => params[:usuario],
'numeroTel' => params[:tel], 'mensTexto' => params[:msg]}

#executando a acao de post dos parametros no servidor de sms
x = Net::HTTP.post_form(URI.parse('http://smsbr.com.br/enviosms.php'), parametros)
  
```

3. Retorno status (HTTP)

PUSH método

O retorno com o status de entrega da mensagem é enviado para uma URL de destino previamente cadastrada na área do cliente na aba Home > Preferências,

Atributo (Variável de retorno)	Descrição
envioData	Data do envio Formato (00/00/0000)
APIMsgID	ID da Mensagem Max- 8 (3ae2d58e) alpha numerico
status	Statual atual da mensagem (001,002,003...) Ver lista
valor	Valor cobrado pelo envio (0.8)

PULL método

http://smsbr.com.br/retorno/statusPull.php?chaveAPI=CHAVE_API&usuarioNome=SEU_LOGIN&APIMsgID=ID_DA_MENSAGEM

Legenda dos status

- 001: Mensagem desconhecida
- 002: Mensagem na fila de entrega
- 003: Entregue ao Gateway
- 004: Recebida pelo celular
- 005: Erro na mensagem
- 006: Cancelada pelo usuário
- 007: Erro na entrega da mensagem
- 008: Recebida pelo gateway
- 009: Erro no encaminhamento
- 010: Mensagem expirou
- 011: Enfileiradas para posterior entrega
- 012: Numero inválido
- 013: Numero de origem inválido
- 014: Rota não disponível

Após 45 dias as mensagens são excluídas do histórico, deste modo o método Pull não pode ser utilizado após este período, o cliente poderá consultar estas mensagens no Relatório de Backup

4. Comando adicionais GET

Consultar créditos disponíveis

http://smsbr.com.br/retorno/retornoCreditos.php?chaveAPI=CHAVE_API&usuarioNome=SEU_LOGIN

5. Consultas HLR

Para efetuar uma consulta HLR, acessar no painel do usuário a ferramenta HLR.

6. Retorno de Envio

Após o envio nossos servidores irão lhe retornar um status inicial de entrega.

Código status

STATUS	VALOR	DESCRIÇÃO
LIMITE CARACTER	Er001	O limite maximo de (150) caracteres aceito foi atingido.
ERRO AUTENTICAÇÃO	Er002	Erro de autenticação.
CREDITOS INSUFICIENTES	Er003	Créditos acabaram.