



Poll via SMS

Objective:

To capture customer feedback through a poll on SMS, after she receives the order placed on an e-commerce website

Implementation:

Prerequisite before using APIs:

1. Sign in to gupshup.io to get an API Key.
2. A Gupshup Enterprise SMS account to send SMS.

Actual implementation:

1. Create a poll smart message that will enable us to take website feedback.

URL Endpoint and HTTP Methods: Creating a Poll

```
PUT : http://api.gupshup.io/sm/api/smartmsg/poll
```

HTTP Request using Curl to create a Poll:

```
curl -X PUT http://api.gupshup.io/sm/api/smartmsg/poll \
-H "apikey: YOUR API KEY" \
-H "Content-Type: application/x-www-form-urlencoded" \
-d 'question=Are you satisfied with the products received in your
order?&callbackUrl=http://requestb.in/se2qvrse'
```

Response body:

```
{
  "id": "19",
  "meta": {
    "callbackurl": "http://requestb.in/se2qvrse",
    "deleted": false,
    "expired": false
  },
  "payload": {
    "content": "Are you satisfied with the products received in your order?",
    "type": "poll"
  }
}
```

2. Once the poll is created, generate a signed link for a smart message and get the embedlink using SignedLink API.

URL Endpoint and HTTP Methods: Generate a signed link for a smart message:

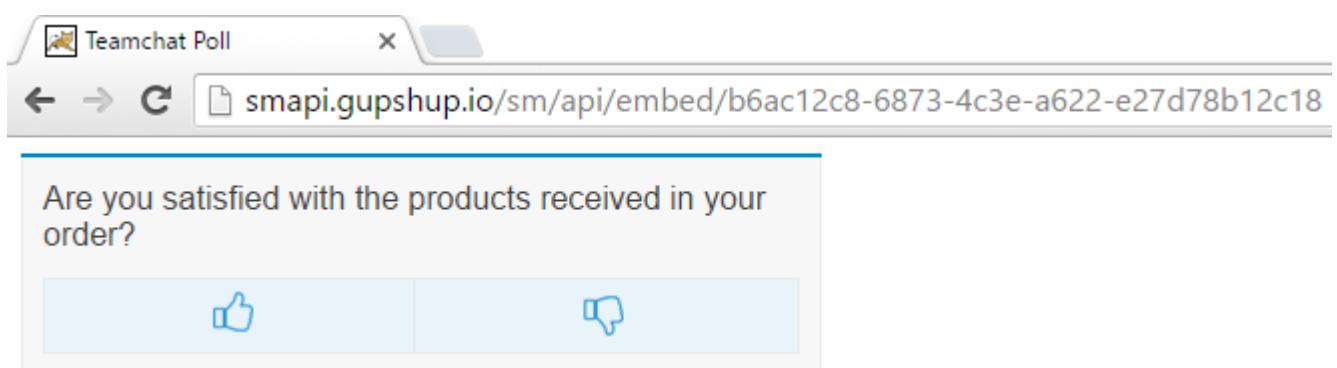
```
POST : http://api.gupshup.io/sm/api/smartmsg/msg/{id}/signedlink
```

HTTP Request to generate a signed-link for user:

```
curl -X POST http://api.gupshup.io/sm/api/smartmsg/msg/19/signedlink \
-H "apikey: YOUR API KEY" \
-H "Content-Type: application/x-www-form-urlencoded" \
-d 'destination = Target user'
```

Response body:

```
[  
  {  
    "embedlink":  
      "http://smapi.gupshup.io/sm/api/embed/b6ac12c8-6873-4c3e-a622-e27d78b12c18",  
    "expired": false,  
    "id": "b6ac12c8-6873-4c3e-a622-e27d78b12c18",  
    "signed-for": {  
      "display": "Target user",  
      "subdisplay": "Target user"  
    },  
    "smid": "19"  
  }  
]
```



3. Finally embed the url in your SMS message body and send an SMS using Send SMS API:

You can convert the long URL into tiny URL using [Gupshup tiny url](#)

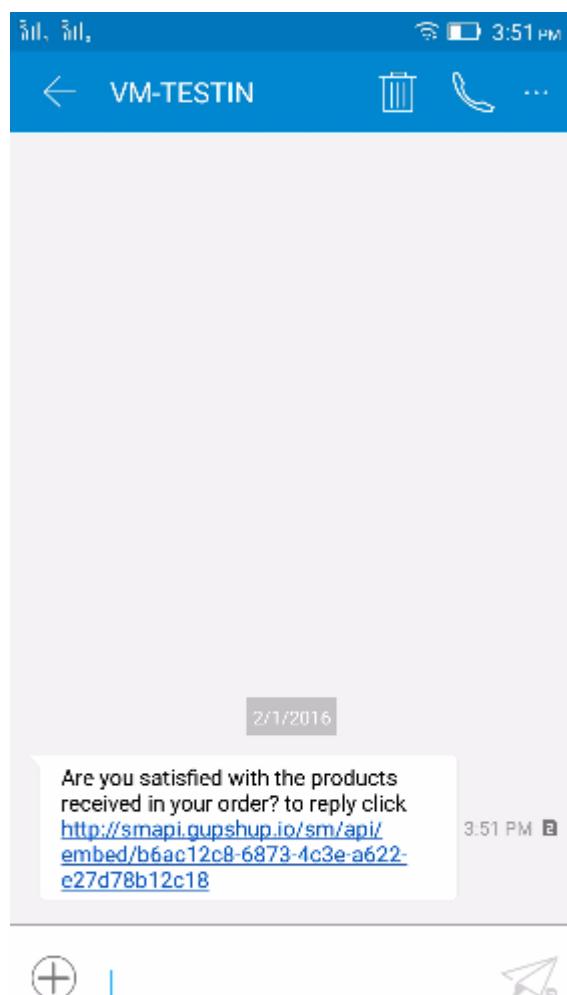
URL Endpoint and HTTP Methods: Send an SMS

```
PUT : http://api.gupshup.io/sm/api/sms/msg
```

HTTP Request using Curl to send an SMS:

```
curl -X PUT http://api.gupshup.io/sm/api/sms/msg \
-H "apikey: YOUR API KEY" \
-d 'destination=PHONE NUMBER \
&text=Are you satisfied with the products received in your order? to reply
click http://smapi.gupshup.io/sm/api/embed/b6ac12c8-6873-4c3e-a622-e27d78b12c18' \
-H "Content-Type: application/x-www-form-urlencoded"
```

SMS OUTPUT.



Response Code: Response code 204 on success

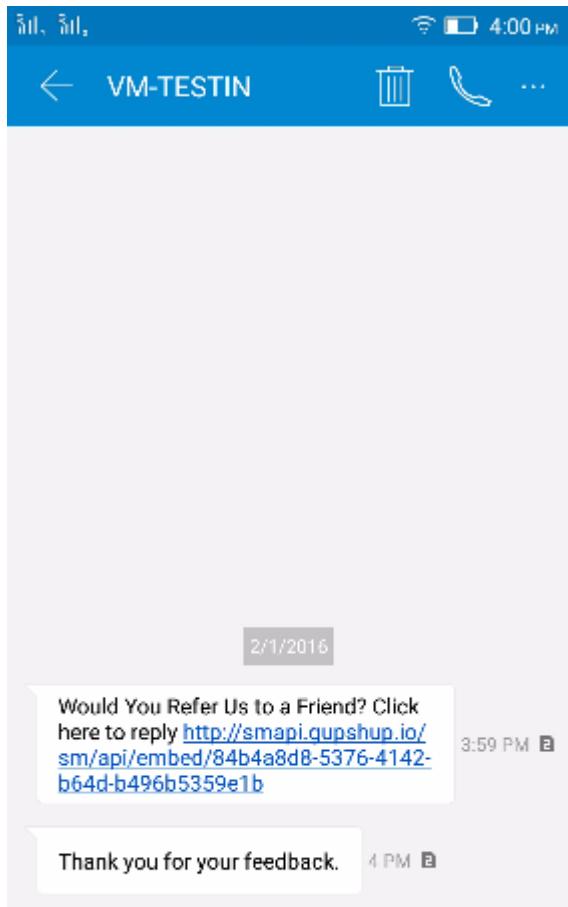
4. Callback: When you select a particular option following parameters are pass to your callback URL in the form:

```
GET : http://requestb.in/se2qvrse?response=yes&user=Target User
```

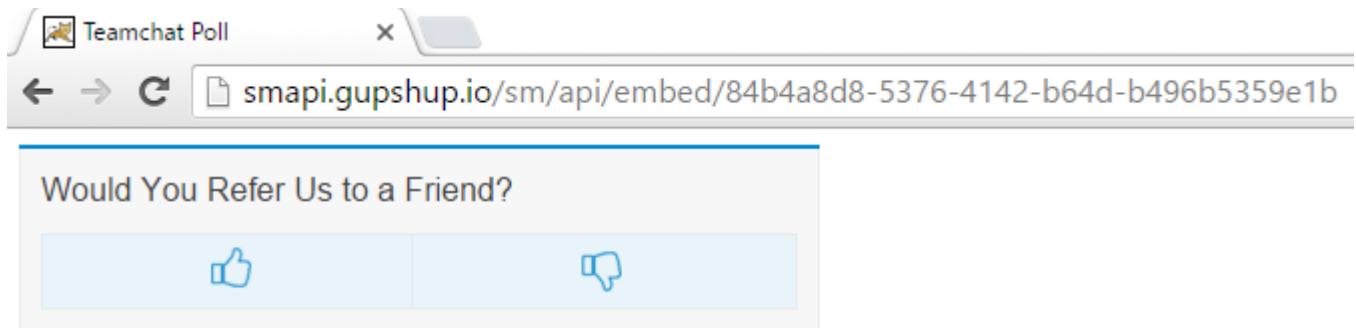
Pseudo code for trigger an SMS based on reply to survey smart message:

```
getParameter("user");
getParameter("response");
String URL = Create a poll using poll API and get embedded URL.
if (response = "yes" || "No")
{
    String smsBody = "Would You Refer Us to a Friend?
                      Click here "+URL;
}
else
{
    String smsBody = "Thank you for your Feedback";
}
4. Make an http Put request to
    "http://api.gupshup.io/sm/api/sms/msg"
    with text = smsBody and destination = phoneNumber as payload.
```

Message Output:



Poll Output



Check out the [example](#) on GitHub.