

SMS4BD Gateway Integration

This document is intended to help IT experts in integration of SMS gateway of SMS4BD. Any alteration of this document without permission is strictly prohibited. This document is a copyright of SMS4BD.

TECHNICAL
DOCUMENTATION
v3.0

CONTENTS

| | |
|---|---|
| Abstract: | 2 |
| Integration Steps: | 3 |
| Integration types: | 3 |
| Outgoing SMS Integration: | 3 |
| Incoming SMS Integration: | 3 |
| Outgoing SMS HTTP endpoints: | 4 |
| Outgoing SMS HTTP Parameters: | 4 |
| Outgoing message delivery status: | 5 |
| Outgoing message delivery status Example: | 6 |
| Incoming Operation endpoints: | 6 |
| Few IMportant Things to note: | 7 |
| HTTP API Example (Sending a single SMS): | 7 |
| HTTP API Example (Checking Balance): | 7 |
| PHP Code Example (Sending a single SMS): | 8 |
| PHP Code Example (Checking Balance): | 8 |
| .Net Code Example (Sending a single SMS): | 8 |
| .Net Code Example (Sending a single SMS - Unicode): | 9 |
| .Net Code Example (Checking Balance): | 9 |

ABSTRACT:

SMS4BD provides SMS gateway that is easily attachable to various platforms. We utilize the power of core HTTP Post request for integration, so naturally anyone from any platform can integrate it with their application. The detail of integration is explained later in this document along with code sample.

INTEGRATION STEPS:

1. Contact with SMS4BD and register an account in our website <http://www.sms4bd.com>.
2. Get the API for your desired platform (contact us to get this or download from our website).
3. Read the documentation to understand the integration process.
4. Host your code to communicate with our server.
5. After approval and payment we will add balance in your account.

INTEGRATION TYPES:

There are two types of integration for gateway.

1. Outgoing (Push service)
2. Incoming (Push pull service)

OUTGOING SMS INTEGRATION:

For outgoing SMS integration the required steps are as follows:

1. Collect your api public key and private key from your panel (Under settings-> Update personal info).
2. Add a masking request from your panel (Under Settings -> Maskings)
3. Wait for masking approval usually we approve within 24 hours but if you need it urgent, then you can contact us immediately. Masking should be within 11 characters and contain only alphanumeric characters.
4. You can buy credit for SMS anytime through bKash or bank transfer.
5. Write a code that uses our provided API for your platform.
6. Send SMS send request.
7. Receive response from the API and process it according to your need in your own application.

INCOMING SMS INTEGRATION:

For incoming SMS integration the required steps are as follows:

1. Add a keyword request from your panel (Under Settings -> Keywords) and there provide a URL where you will receive the SMS.
2. It may take up to 7 days to approve your keyword based on availability.
3. When we approve the keyword you can receive the SMS whenever some send SMS to 6969 using your keyword.

OUTGOING SMS HTTP ENDPOINTS:

The base address for the service is: <http://www.sms4bd.net/http.svc/SendSMS>

You can use HTTP GET calls to our endpoint to consume our web service. This is useful when you do not want to use our API or we do not provide API for your programming language platform. In this case you can use bare bone HTTP API to integrate and consume the service. Please make sure to use content type: `"application/x-www-form-urlencoded"`

OUTGOING SMS HTTP PARAMETERS:

The following table contains the HTTP parameters that we accept in our web service. If you use our API you don't need to use these though, as it will be handled by our API automatically.

| Name | Signature | Data Type | Explanation |
|------------------|-------------------------|-----------|---|
| Public Key | <code>publickey</code> | String | Your account Public Key, found in your panel under settings |
| Private Key | <code>privatekey</code> | String | Your account Private Key, found in your panel under settings |
| Sender / Masking | <code>sender</code> | String | The masking to be used. This masking needs to be preapproved. Apply for masking from panel under settings |
| Receiver Mobile | <code>receiver</code> | String | Mobile number to send SMS. Should start with country code. You can use + sign or can use the country code only. (i.e. +8801717570000) |
| Message Body | <code>message</code> | String | The SMS body that you want to send |
| Delay In Seconds | <code>delay</code> | Integer | If you want to schedule the SMS for later delivery then set this |

| | | | |
|-------------|------|---------|--|
| | | | value to total seconds that you want to delay the SMS. Do not use too large value to avoid overflow. |
| Type Of SMS | type | Integer | The type of the SMS. For ASCII (Plain English SMS) set 1, for Unicode SMS set 8 |

OUTGOING MESSAGE DELIVERY STATUS:

You can use the following codes to determine the status of the sent SMS. After you make SendSMS call, you to our HTTP endpoint, you will receive a string in return which may contain 2 or 3 elements separated by comma. If the Status of the SMS submission is Accepted then it will contain 3 elements, status code, status message and the reference id of the submission for future investigation or delivery report reception. If the status code is Rejected or anything else than Accepted, then there will be only 2 elements, the status code and the status message as the reference id is not available due to failure. Following is the table for various status codes.

| Code | Message | Explanation |
|----------|-----------|---|
| 0 | None | No status was decided yet. |
| 1 | Enroute | The SMS was handed over to SMSC and waiting for delivery. Usually if there is long delay due to network traffic, this status can be received. In normal condition, you may not receive this status due to quick processing of SMS |
| 2 | Delivered | The message was successfully delivered to mobile. |
| 3 | Expired | It was not possible to deliver the message to recipient. Possible reasons can be recipient out of network, turned off phone for long time or discontinued SIM |
| 4 | Deleted | The SMS was deleted from delivery. Possible reasons can be government policy change or instruction from regulatory board to do so. Can be applied during emergency situation. |

| | | |
|----------|---------------|--|
| 5 | Undeliverable | The network is unavailable for SMS transmission. |
| 6 | Accepted | We have received your SMS and submitted it to SMSC for delivery. |
| 7 | Unknown | Due to network error we couldn't determine the status. |
| 8 | Rejected | Your submission was rejected due to invalid data. It can be due to invalid username/password/masking/message. It can also be applied if your balance is low or recipient mobile number is invalid. |

OUTGOING MESSAGE DELIVERY STATUS EXAMPLE:

Successful response:

6,Accepted,74461004-52c2-cad0-07bd-08d14e9f13df

Failed response:

8,Rejected

INCOMING OPERATION ENDPOINTS:

For incoming SMS you have to provide us a keyword (3 to 4 alphanumeric characters) and a hit URL that we will call to get the return message from your end. We will send the reply to the Sender mobile automatically based on the response received from your hit URL.

For example, if you provide us a keyword (if it is available): xyz

And you provide us a URL: www.example.com/sendreply.php,

If any user sends an SMS to 6969 with the following message: "xyz balance 18383"

We will hit your above url with the following post parameters: mobile, message.

Here mobile will be: senders mobile number (for example, 8801758783...), message will be: "xyz balance 18383".

Please make sure that you have applied for the keyword from your panel under settings and it has been approved. We need around 7 working days to set this up for you. Also the keyword is subject to availability.

Please feel free to contact us to get information if you face difficulty.

FEW IMPORTANT THINGS TO NOTE:

For the gateway requests, all date times must be in UTC+0. If there is an API SDK available for your programming language then please use it instead of manual implementation as it can save lot of your time and efforts.

HTTP API EXAMPLE (SENDING A SINGLE SMS):

HTTP API is very quick and easy. The easiest and fastest way to send SMS will be to use the following URL format: You need to make a HTTP GET request in this URL.

You need to replace {1}, {2}, {3}, {4}, {5}, {6}, {7} etc with actual value as explained in the above table in outgoing SMS HTTP parameter list.

<http://www.sms4bd.net/HTTP.svc/SendSMS?publickey={1}&privatekey={2}&sender={3}&receiver={4}&message={5}&delay={6}&type={7}>

HTTP API EXAMPLE (CHECKING BALANCE):

HTTP API is very quick and easy. The easiest and fastest way to check balance will be to use the following URL format: You need to make a HTTP GET request in this URL.

You need to replace {1}, {2} etc with actual value as explained in the above table in outgoing SMS HTTP parameter list.

<http://www.sms4bd.net/HTTP.svc/SendSMS?publickey={1}&privatekey={2}>

PHP CODE EXAMPLE (SENDING A SINGLE SMS):

```
<?php
namespace HttpSDK_SMS;
require_once 'SMSClient.php';

$client = new SMSClient("publickey", "privatekey", "http://www.sms4bd.net");
$response = $client->SendSMS("SMS4BD", "8801717570000",
    "This is PHP", date('Y-m-d H:i:s'), SMSType::ASCII);

echo $response->StatusMessage;
```

PHP CODE EXAMPLE (CHECKING BALANCE):

```
<?php
namespace HttpSDK_SMS;
require_once 'SMSClient.php';

$client = new SMSClient("publickey", "privatekey", "http://www.sms4bd.net");
$response = $client->GetBalance();

echo $response;
```

.NET CODE EXAMPLE (SENDING A SINGLE SMS):

```
using System;
using HttpSDK.SMS;

namespace SMS4BD.API.Tests
{
    public class NewAPITests
    {
        public void SendSMSTest()
        {
            SMSClient client = new SMSClient("publickey", "privatekey",
                "http://www.sms4bd.net");

            SMSResponse result = client.SendSMS("SMS4BD", "8801717570000",
                "This is a test message.",
                DateTime.UtcNow, SMSType.ASCII);

            if (result.StatusCode == 6)
            {
                Console.WriteLine("Success:" + result.StatusMessage);
            }
            else
            {
                Console.WriteLine("Fail:" + result.StatusMessage);
            }
        }
    }
}
```

.NET CODE EXAMPLE (SENDING A SINGLE SMS - UNICODE):

```
using System;
using HttpSDK.SMS;

namespace SMS4BD.API.Tests
{
    public class NewAPITests
    {
        public void SendSMSTest()
        {
            SMSClient client = new SMSClient("publickey", "privatekey",
                "http://www.sms4bd.net");

            SMSResponse result = client.SendSMS("SMS4BD", "8801717570000",
                "শাকিন আহমেদ লিখনের বয়স ১৩ বছর।",
                DateTime.UtcNow, SMSType.UCS2);

            if (result.StatusCode == 6)
            {
                Console.WriteLine("Success:" + result.StatusMessage);
            }
            else
            {
                Console.WriteLine("Fail:" + result.StatusMessage);
            }
        }
    }
}
```

.NET CODE EXAMPLE (CHECKING BALANCE):

```
using System;
using HttpSDK.SMS;

namespace SMS4BD.API.Tests
{
    public class NewAPITests
    {
        public void GetBalanceTest()
        {
            SMSClient client = new SMSClient("publickey", "privatekey",
                "http://www.sms4bd.net");

            double result = client.GetBalance();

            Console.WriteLine(result);
        }
    }
}
```

Please contact info@sms4bd.com, or call +88 01833 316 571, +88 02 9822167 to consult and resolve integration issues. We are happy to help you.