



Interfejs HTTP/S API

Wersja 2.34



SPIS TREŚCI

1. Wprowadzenie

- 1.1 Rozpoczęcie współpracy
- 1.2 Filtr IP dla interfejsu API
- 1.3 Hasło do interfejsu API

I. WYSYŁKA SMS

2. Wysyłanie pojedynczych SMS'ów

- 2.1 Wysyłanie wiadomości FAST
- 2.2 Wysyłanie SMS-a o zaplanowanej godzinie/dacie
- 2.3 Usuwanie zaplanowanej wiadomości

3. Wysyłanie masowe SMS-ów

- 3.1 Wysyłanie SMS do grupy z książki telefonicznej
- 3.2 Wysyłanie masowe spersonalizowanych wiadomości z wykorzystaniem parametrów
- 3.3 Wysyłanie masowe wiadomości z wykorzystaniem parametru IDX
- 3.4 Wysyłanie wiadomości z wykorzystaniem Kodów rabatowych
- 3.5 Wysyłanie wiadomości z wykorzystaniem idz.do

4. Wysyłanie wiadomości z wykorzystaniem szablonów

5. mail2SMS – Wysyłanie SMS za pomocą e-mail

6. Sprawdzenie ilości punktów na koncie

II. WYSYŁKA MMS

7. Wysyłanie wiadomości MMS

III. WYSYŁKA VMS

8. Wysyłanie wiadomości VMS

IV. SPRAWDZANIE RAPORTÓW DORĘCZENIA WIADOMOŚCI

- 9. Odbieranie raportu doręczenia wiadomości SMS - CALLBACK
- 10. Odbieranie raportu doręczenia wiadomości MMS - CALLBACK
- 11. Odbieranie raportu doręczenia wiadomości VMS - CALLBACK
- 12. CALLBACK dla wysyłek masowych
- 13. CALLBACK kliknięć idz.do

V. ODBIÓR WIADOMOŚCI SMS ORAZ MMS

14. Odbiór SMS/MMS

- 14.1 Odbiór SMS
- 14.2 Odbiór MMS



VI. HLR – System sprawdzania numeru

15. HLR – sprawdzanie poprawności numeru

VII. PODSUMOWANIE

16. Uwagi końcowe

Dodatek nr 1 – Lista statusów doręczenia wiadomości

Dodatek nr 2 – Kody błędów

Dodatek nr 3 – Kodowanie

Dodatek nr 4 – Przykładowe skrypty

Historia zmian



1. Wprowadzenie

Platforma SMSAPI została skierowana do użytkowników chcących rozbudować swoje aplikacje o system wysyłania i odbierania SMS-ów oraz MMS-ów oraz wysyłki wiadomości głosowych VMS. Aplikacja ta w prosty sposób umożliwia integrację dowolnego serwisu z bramką SMS, MMS i/lub VMS. Głównymi atutami naszego serwisu oprócz prostej implementacji jest możliwość nadawania wiadomości z własnej nazwy (maksymalnie 11 znaków). Każdy wysłany SMS, MMS oraz VMS za pośrednictwem systemu posiada unikalny numer identyfikacyjny pozwalając na sprawdzenie raportu doręczenia wiadomości.

1.1 Rozpoczęcie współpracy

W celu rozpoczęcia współpracy należy utworzyć konto w serwisie SMSAPI: <http://www.smsapi.pl/> Utworzone konto jest gotowe do użytku, jednak zalecamy ustawienie własnego pola nadawcy. Jako domyślne ustawione jest pole nadawcy „Info”. Dodanie pola nadawcy jest usługą całkowicie darmową.

1.2 Filtr IP dla interfejsu API

W celu dodatkowego zabezpieczenia interfejsu API w menu „Ustawienia API” → „Filtr adresów IP” w polu **Filtr adresów IP** ustawić można adresy IP z których możliwa będzie wysyłka wiadomości (w przypadku próby dokonania wysyłki z innego IP system zwróci błąd: ERROR:105). Adresy należy oddzielić średnikami.

1.3 Hasło do interfejsu API

Hasło do interfejsu API po rejestracji jest takie same jak hasło do panelu WWW. Istnieje możliwość zmiany hasła do interfejsu API w zakładce **Ustawienia API** → **Hasło do API**. Zmiana hasła do panelu WWW nie wpływa na hasło API.

2. Wysyłanie pojedynczych SMS'ów

Adresy URL do połączenia z aplikacją – zwane dalej „Adresem połączenia”:

- <https://api.smsapi.pl/sms.do> - dla połączeń szyfrowanych SSL
- <https://api2.smsapi.pl/sms.do> - backup dla połączeń szyfrowanych SSL
- <http://api.smsapi.pl/sms.do> - dla połączeń standardowych (**Niezalecane!**)
- <http://api2.smsapi.pl/sms.do> - backup dla połączeń standardowych (**Niezalecane!**)

Wysyłanie SMS-ów odbywa się przez wysłanie metodą **GET** lub **POST** danych do adresu połączenia:

Parametr	Opis
<i>username</i> *	Nazwa użytkownika lub główny adres e-mail przypisany do konta w serwisie SMSAPI
<i>password</i> *	Hasło do API które można zmienić w zakładce Ustawienia API → Hasło do API . Po rejestracji hasła do API i do Panelu są takie same. Hasło powinno być zakodowane w md5.
<i>to</i>	Numer odbiorcy wiadomości w formacie 48xxxxxxxx lub xxxxxxxx. Np. 48505602702 lub 505602702.
<i>group</i>	Nazwa grupy kontaktów z książki telefonicznej, do których ma zostać wysłana wiadomość
<i>message</i> *	Treść wiadomości. Standardowo do 160 znaków lub 70 znaków w przypadku wystąpienia chociaż jednego znaku specjalnego (polskie znaki uważane są za specjalne). Maksymalna długość wiadomości wynosi 918 znaków (lub 402 ze znakami specjalnymi) i jest wysyłana/ jako 6 połączonych SMS-ów, obciążając konto zgodnie z aktualnym cennikiem. Więcej szczegółów odnośnie znaków specjalnych znajduje się w pkt 16.
<i>from</i>	Parametr <i>from</i> definiuje rodzaj wysłanej wiadomości (Pro / Eco / 2way) jednocześnie dla wiadomości Pro określając Pole Nadawcy z jakim wiadomość ma być wysłana: <ul style="list-style-type: none"> - <i>&from=Nazwa</i> – spowoduje wysłanie wiadomości Pro z polem nadawcy „Nazwa” - <i>&from=2way</i> – spowoduje wysłanie wiadomości 2WAY - <i>&from=Eco</i> – spowoduje wysłanie wiadomości ECO Pozostawienie pola pustego powoduje wysłanie domyślnego rodzaju wiadomości z domyślnym polem nadawcy. Przyjmowane są tylko nazwy zweryfikowane. (&from=aktywna_nazwa). Pole nadawcy należy dodać po zalogowaniu na stronie SMSAPI, w zakładce Wiadomości SMS → Pola nadawcy .
<i>encoding</i>	Parametr określa kodowanie polskich znaków w SMS-ie. Domyślne kodowanie jest windows-1250. Jeżeli występuje konieczność zmiany kodowania, należy użyć parametru encoding z danymi:



	<ul style="list-style-type: none"> - dla iso-8859-2 (latin2) – należy podać wartość „iso-8859-2”, - dla utf-8 – należy podać wartość „utf-8”.
<i>flash</i>	Wysyłanie wiadomości trybem „flash”, odbywa się poprzez podanie parametru flash o wartości „1”. SMS-y flash są automatycznie wyświetlane na ekranie głównym telefonu komórkowego i nie są przechowywane w skrzynce odbiorczej (jeśli nie zostaną zapisane).
<i>test</i>	Wiadomość nie jest wysyłana, wyświetlana jest jedynie odpowiedź (w celach testowych). (&test=1)
<i>details</i>	W odpowiedzi zawarte jest więcej szczegółów. (Treść wiadomości, długość wiadomości, ilość części z jakich składa się wiadomość). (&details=1)
<i>date</i>	Data w formacie unixtime (&date=1287734110) lub ISO 8601 (&date=2012-05-10T08:40:27+00:00). Określa kiedy wiadomość ma być wysłana. W przypadku wstawienia daty przeszłej wiadomość zostanie wysłana od razu. Wiadomość można zaplanować na maksymalnie 3 miesiące do przodu.
<i>date_validate</i>	Ustawienie „1” sprawdza poprawność formatu podanej daty. W przypadku wystąpienia błędnej daty zwrócony zostanie błąd ERROR:54
<i>datacoding</i>	Parametr pozwalający na wysyłanie wiadomości binarnych. (&datacoding=bin) Parametry udh oraz message muszą być wysyłane w postaci ciągu HEX (np. message=616263 dla treści abc)
<i>udh</i>	Nagłówek UDH wiadomości SMS wymagany podczas korzystania z parametru &datacoding=bin (np. &udh=0605040b8423f0 dla WAP PUSH)
<i>skip_foreign</i>	Ustawienie tego parametru (&skip_foreign=1) powoduje pominięcie numerów nie-polskich jeżeli takie pojawią się w odwołaniu.
<i>allow_duplicates</i>	pozwala na wysłanie wiadomości na numery zduplikowane w danym odwołaniu (przydatne np. przy wykorzystywaniu sparametryzowanych treści)
<i>idx</i>	Opcjonalny parametr użytkownika wysyłany z wiadomością a następnie zwracany przy wywołaniu zwrotnym CALLBACK. Parametr idx może mieć maksymalnie 255 znaków dopuszczalne są cyfry 0 - 9 oraz litery a – z (wielkość liter nie jest rozróżniana). (&idx=123)
<i>check_idx</i>	Pozwala zabezpieczyć przed wysłaniem dwóch wiadomości z identyczną wartością parametru <i>idx</i> w trakcie ostatnich 24h. W przypadku ustawienia parametru (&check_idx=1) system sprawdza czy wiadomość z podanym idx nie została wysłana w ostatnich czterech dniach lub czy nie jest zaplanowana do wysłania w przyszłości, jeżeli jest zwracany jest błąd 53.
<i>nounicode</i>	Ustawienie 1 zabezpiecza przed wysłaniem wiadomości ze znakami specjalnymi (w tym polskimi) (ERROR:11).
<i>normalize</i>	Opcja spowoduje zastąpienie w wiadomości SMS znaków specjalnych na ich odpowiedniki (np. ą-a, ć-c, ę-e...). (&normalize=1)
<i>fast</i>	Ustawienie 1 spowoduje wysłanie wiadomości przy wykorzystaniu osobnego kanału zapewniającego szybkie doręczenie wiadomości Fast. Z parametru korzystać można podczas wysyłania wiadomości Pro oraz Eco, ilość punktów za wysyłkę pomnożona będzie przez 1.5 Uwaga! Dla tego parametru zabronione jest prowadzenie wysyłek masowych i marketingowych.
<i>partner_id</i>	Kod partnerski, który otrzymać można po podpisaniu umowy partnerskiej. Kod nie będzie brany pod uwagę jeżeli użytkownik wysyłający polecony jest przez innego klienta lub podaje swój kod.
<i>max_parts</i>	Parametr określa maksymalną ilość części z jakich może składać się wiadomość. Maksymalna wartość to 6. W przypadku gdy wiadomość składa się z większej ilości części zwrócony zostanie błąd ERROR:12 Domyślna ilość części ustawiana jest w panelu klienta.
<i>expiration_date</i>	Różnica pomiędzy datą wysyłki a datą wygaśnięcia musi być większa niż 15 minut i mniejsza niż 72 godzin (zaleca się aby było to minimum 1 godzina i maksymalnie 12 godzin). Data może zostać podana w formacie unixtime (&date=1287734110) lub ISO 8601 (&date=2012-05-10T08:40:27+00:00).
<i>discount_group</i>	Nazwa grupy kodów rabatowych do załączenia w wysyłce. Grupy można dodawać oraz edytować w panelu klienta.
<i>notify_url</i>	Parametr pozwalający na ustawienie adresu URL do skryptu callback odbierającego raporty doręczenia dla danej wiadomości. Adres ten jest brany w pierwszej kolejności (przed globalnym adresem ustawionym na koncie).
<i>format</i>	Dla &format=json powoduje, że zwrotka z API wysyłana jest w formacie JSON.



* - pole obowiązkowe

UWAGA! Parametry **group** oraz **to** są zamienne, w odwołaniu musi się pojawić jeden z tych dwóch parametrów. Brak jednego z tych dwóch parametrów lub wystąpienie ich obu spowoduje niewysłanie wiadomości oraz zwrócenie błędu ERROR: 13.

Zapytanie: **https://api.smsapi.pl/sms.do?**
**username=uzytkownik&password=hasloMD5&from=nazwa&to=48100200300
&message=test wiadomosci&format=json**

I. Przykłady odpowiedzi z użyciem parametru **&format=json**:

a) w przypadku powodzenia:

```
{
  "count": 1,
  "list": [
    {
      "id": "1460969715572091219", //id wiadomości
      "points": 0.16, //koszt wysyłki
      "number": "48100200300", //numer tel. odbiorcy z kierunkowym krajem
      "date_sent": 1460969712, //data wysyłki
      "submitted_number": "48100200300", //numer tel. wysłany w odwołaniu
      "status": "QUEUE" //status wiadomości
    }
  ]
}
```

b) w przypadku błędu:

```
{
  "invalid_numbers": [
    {
      "number": "48100200300", //numer tel. odbiorcy z kierunkowym krajem
      "submitted_number": "48100200300", //numer tel. wysłany w odwołaniu
      "message": "Invalid phone number" //opis błędu
    }
  ],
  "error": 13, //kod błędu
  "message": "No correct phone numbers" //opis błędu
}
```

II. Przykłady odpowiedzi bez użycia parametru **&format**:

a) w przypadku powodzenia:

OK:<id>:<points>

b) w przypadku błędu:

ERROR:<err>

Gdzie:

- <id>** numer ID wysłanego SMS-a. Numer należy zachować w celu sprawdzenia statusu wiadomości. Parametr ten może być zmiennej długości, nieprzekraczającej 32 znaki.
- <points>** ilość pobranych punktów (np. SMS składający się z 3 części zwróci wartość 3 x (stawka SMS)).
- <err>** kod błędu (zobacz dodatek – kody błędów)



Przykład odwołania z żądaniem wysłania wiadomości **PRO**:

[https://api.smsapi.pl/sms.do?username=uzytownik&password=hasloMD5&from=nazwa&to=48500500500&message=test wiadomosci_PRO&format=json](https://api.smsapi.pl/sms.do?username=uzytownik&password=hasloMD5&from=nazwa&to=48500500500&message=test%20wiadomosci_PRO&format=json)

Przykład odwołania z żądaniem wysłania wiadomości **ECO**:

[https://api.smsapi.pl/sms.do?username=uzytownik&password=hasloMD5&from=ECO&to=48500500500&message=test wiadomosci_ECO&format=json](https://api.smsapi.pl/sms.do?username=uzytownik&password=hasloMD5&from=ECO&to=48500500500&message=test%20wiadomosci_ECO&format=json)

Przykład odwołania z żądaniem wysłania wiadomości **2WAY**:

[https://api.smsapi.pl/sms.do?username=uzytownik&password=hasloMD5&from=2WAY&to=48500500500&message=test wiadomosci_2WAY&format=json](https://api.smsapi.pl/sms.do?username=uzytownik&password=hasloMD5&from=2WAY&to=48500500500&message=test%20wiadomosci_2WAY&format=json)

W razie problemów podczas odwoływania się do podstawowego adresu URL (jak w przykładzie powyżej) można wykorzystać adres URL backup **<https://api2.smsapi.pl/>**

2.1 Wysyłanie wiadomości FAST

Ustawienie parametru fast=1 spowoduje wysłanie wiadomości przy wykorzystaniu osobnego kanału zapewniającego najwyższy priorytet wysyłki. Z parametru korzystać można podczas wysyłania wiadomości Pro oraz Eco, ilość punktów za wysyłkę pomnożona będzie przez 1.5

Zapytanie: **[https://api.smsapi.pl/sms.do?username=uzytownik&password=hasloMD5&format=json&to=48500500500&fast=1&message=test wiadomosci Fast](https://api.smsapi.pl/sms.do?username=uzytownik&password=hasloMD5&format=json&to=48500500500&fast=1&message=test%20wiadomosci_Fast)**

UWAGA! Dla tego parametru nie ma możliwości wysyłania wiadomości masowych i marketingowych. W przypadku wysyłki do więcej niż jednego numeru w jednym odwołaniu wiadomości zostaną wysłane jako zwykłe wiadomości.

2.2 Wysyłanie SMS-a o zaplanowanej godzinie/dacie

W celu wysłania wiadomości o określonej godzinie/dacie musi zostać użyty parametr **date** wskazujący datę, którą należy skonwertować do postaci unixtime.

Zapytanie: **[https://api.smsapi.pl/sms.do?username=uzytownik&password=hasloMD5&to=48500500500&date=1577878200&message=wiadomosc zaplanowana&format=json](https://api.smsapi.pl/sms.do?username=uzytownik&password=hasloMD5&to=48500500500&date=1577878200&message=wiadomosc%20zaplanowana&format=json)**



2.3 Usuwanie zaplanowanej wiadomości

Parametr	Opis
<i>username</i> *	Nazwa użytkownika lub główny adres e-mail przypisany do konta w serwisie SMSAPI
<i>password</i> *	Hasło do Twojego konta w serwisie SMSAPI zaszyfrowane w MD5
<i>sch_del</i> *	ID zaplanowanej wiadomości (zwrócony podczas wysyłki), w przypadku większej liczby zaplanowanych wiadomości do usunięcia należy podawać kolejne ID po przecinku.

* - pole obowiązkowe

Zapytanie: **[https://api.smsapi.pl/sms.do?username=uzytownik&password=haslo
&sch_del=09040616088106874&format=json](https://api.smsapi.pl/sms.do?username=uzytownik&password=haslo&sch_del=09040616088106874&format=json)**

I. Przykłady odpowiedzi z użyciem parametru **&format=json**:

a) w przypadku powodzenia:

```
{
  "count": 1, //liczba usuwanych wiadomości
  "list": [
    {
      "id": "1462974172199599210" //id usuwanej wiadomości
    }
  ]
}
```

b) w przypadku błędu:

```
{
  "error": 301, //kod błędu
  "message": "Not exists ID message" //opis błędu
}
```

II. Przykłady odpowiedzi bez użycia parametru **&format**:

a) w przypadku powodzenia:

```
OK:<id>:<points>
```

b) w przypadku błędu:

```
ERROR:<err>
```

W razie problemów podczas odwoływania się do podstawowego adresu URL (jak w przykładzie powyżej) można wykorzystać adres URL backup **<https://api2.smsapi.pl/>**.



3. Wysyłanie masowe SMS-ów

Wysyłanie SMS-ów do grupy odbiorców odbywa się tak jak do pojedynczego odbiorcy (opis w punkcie 2) lecz z podaniem wielu numerów telefonów w parametrze „to”.

Dodatkowo cała wysyłka powinna odbywać się metodą **POST** do adresu połączenia. W przypadku wysyłki metodą **GET** przy większej ilości numerów, część parametrów może zostać ucięta i w efekcie niedostarczona w całości do serwisu SMSAPI.

Jeżeli cena za wysłanie wszystkich SMS-ów jest większa od ilości punktów dostępnych w serwisie, zostanie zwrócony komunikat o błędzie (**103**) i SMS-y nie zostaną wysłane. Jeżeli zostały podane błędne numery (nierozpoznane przez serwis SMSAPI ze względu na błędny prefiks), wszystkie SMS-y zostaną wysłane z pominięciem tych numerów, wówczas ilość otrzymanych raportów będzie różnić się od ilości wysłanych SMS-ów. Sytuacja ta nie dotyczy numerów z poprawnym prefiksem (np. 500000000), gdzie SMS zostanie wysłany, a następnie odrzucony gdyż numer nie istnieje (punkty za wysłanie SMS-a zostaną pobrane). Numery, które się powtarzają zostaną wysłane tylko 1 raz.

Zapytanie: **`https://api.smsapi.pl/sms.do?username=uzytownik&password=hasloMD5&from=nazwa&to=48500500500,48501501501,48502502502&message=test wiadomosci&format=json`**

I. Przykłady odpowiedzi z użyciem parametru **&format=json**:

a) w przypadku powodzenia:

```
{
  "count": 3,
  "list": [
    {
      "id": "1460978572913968440",
      "points": 0.16,
      "number": "48500500500",
      "date_sent": 1460978579,
      "submitted_number": "48500500500",
      "status": "QUEUE"
    },
    {
      "id": "1460978572913968450",
      "points": 0.16,
      "number": "48501501501",
      "date_sent": 1460978579,
      "submitted_number": "48501501501",
      "status": "QUEUE"
    },
    {
      "id": "1460978572913968460",
      "points": 0.16,
      "number": "48502502502",
      "date_sent": 1460978579,
      "submitted_number": "48502502502",
      "status": "QUEUE"
    }
  ]
}
```



b) w przypadku błędu:

```
{
  "invalid_numbers": [
    {
      "number": "456456456",
      "submitted_number": "456456456",
      "message": "Invalid phone number"
    },
    {
      "number": "321321321",
      "submitted_number": "321321321",
      "message": "Invalid phone number"
    }
  ],
  "error": 13,
  "message": "No correct phone numbers"
}
```

II. Przykłady odpowiedzi bez użycia parametru **&format**:

a) w przypadku powodzenia:

```
OK:<id1><points1><phone1>;<id2><points2><phone2>
```

b) w przypadku błędu:

```
ERROR:<err>
```

- | | |
|-----------------------|--|
| <id> | numer ID wysłanego SMS-a. Numer należy zachować w celu sprawdzenia statusu wiadomości. Parametr ten może być zmiennej długości, nieprzekraczającej 32 znaki. |
| <points> | ilość pobranych punktów (np. SMS składający się z 3 części zwróci wartość 3 x (stawka SMS)). |
| <phone> | numer telefonu odbiorcy |
| <err> | kod błędu (zobacz dodatek – kody błędów) |

Warto zauważyć, że przy wysyłce SMS-ów do grupy odbiorców dodatkowo zwracany jest numer telefonu oraz każdy SMS jest zakończany średnikiem (oprócz ostatniego. Nie dotyczy wysyłki z parametrem **&format=json**).

Zalecana maksymalna ilość jednorazowej wysyłki (jedno wywołanie) wynosi **10 000** wiadomości metodą **POST** oraz do **200** wiadomości metodą **GET**.

W razie problemów podczas odwoływania się do podstawowego adresu URL (jak w przykładzie powyżej) można wykorzystać adres URL backup **https://api2.smsapi.pl/**



3.1 Wysyłanie SMS do grupy z książki telefonicznej

W celu wysłania wiadomości do określonej grupy należy w pierwszej kolejności stworzyć ją w panelu WWW w zakładce Baza kontaktów→Grupy.

Istnieje możliwość wstawienia pola własnego do treści wiadomości. Pola własne mogą być definiowane w zakładce Baza kontaktów → Pola własne. Aby wstawić pole, w treści wiadomości należy wpisać [%kontakt.nazwa_pola%]. Podczas wysyłki pole zostanie zastąpione odpowiednią wartością przypisaną do danego kontaktu. Istnieje także możliwość użycia standardowych pól, takich jak:

Parametr	Opis
[%imie%]	Imię przypisane do kontaktu
[%imie_w%]	Wołacz imienia przypisanego do kontaktu
[%nazwisko%]	Nazwisko przypisane do kontaktu
[%kontakt.kraj%]	Kraj
[%kontakt.opis%]	Opis kontaktu
[%kontakt.email%]	Email
[%kontakt.urzadzenie%]	Urządzenie użytkownika (przypisywany automatycznie w przypadku kliknięcia odbiorcy w skrócony link)
[%kontakt.miejscowosc%]	Miejscowość
[%kontakt.przegladarka%]	Przeglądarka (przypisywana automatycznie w przypadku kliknięcia odbiorcy w skrócony link)
[%kontakt.system operacyjny%]	System operacyjny użytkownika (przypisywany automatycznie w przypadku kliknięcia odbiorcy w skrócony link)
[%kontakt.nazwa_pola%]	Niestandardowe pole własne (definiowane w zakładce Baza kontaktów→ Pola własne)

Poniżej przedstawiamy przykład wysyłki do grupy:

Zapytanie: **[https://api.smsapi.pl/sms.do?username=uzytownik&password=hasloMD5&group=Testowa_grupa&message=wiadomosc_wyslana_do_grupy_z_ksiazki_telefonicznej_-_przykladowe_pole_wlansie:\[%kontakt.nazwa_pola%\]](https://api.smsapi.pl/sms.do?username=uzytownik&password=hasloMD5&group=Testowa_grupa&message=wiadomosc_wyslana_do_grupy_z_ksiazki_telefonicznej_-_przykladowe_pole_wlansie:[%kontakt.nazwa_pola%])**

W razie problemów podczas odwoływania się do podstawowego adresu URL (jak w przykładzie powyżej) można wykorzystać adres URL backup **<https://api2.smsapi.pl/>**.



3.2 Wysłanie masowe spersonalizowanych wiadomości z wykorzystaniem parametrów

Istnieje możliwość wysłania do 100 spersonalizowanych wiadomości przy pomocy jednego wywołania wykorzystując parametry. W razie potrzeby wysłania większej ilości wiadomości można wywołać równoległe kilka odwołań.

Parametry powinny być zdefiniowane w wywołaniu jako **param1**, **param2**, **param3**, **param4**, które zastępowały będą [%1%], [%2%], [%3%] oraz [%4%] w treści wiadomości. Wartości parametrów muszą być oddzielone znakiem pipe „|” według wzoru:

param1=Ania|Michał|Andrzej¶m2=Nowak|Kowalski|Nowakowski

Liczba parametrów musi być identyczna z liczbą numerów do których wysłane mają być wiadomości w innym przypadku zwrócony będzie błąd: ERROR:18 i wiadomość nie zostanie wysłana.

WAŻNE! Długość wiadomości może być różna w zależności od długości parametru!

Jeżeli jeden z numerów będzie błędny zostanie on pominięty i wiadomości zostaną wysłane z pominięciem tego numeru.

Parametry

Parametry wykorzystane mogą być po zdefiniowaniu ich miejsca w wiadomości:

Parametr	Wartość
[%1%]	Tekst parametru 1 (param1)
[%2%]	Tekst parametru 2 (param2)
[%3%]	Tekst parametru 3 (param3)
[%4%]	Tekst parametru 4 (param4)

Przykład:

```
https://api.smsapi.pl/sms.do?username=login&password=hasloMD5&from=nadawca
&to=48600111222,48500111222&message=Test wiadomosci, parametr1: [%1%] parametr2: [%2%]
&param1=Jan|Ania&param2=30|40
```

Wiadomości będą miały następującą treść:

Wiadomość 1: *Test wiadomosci, parametr1: Jan parametr2: 30*

Wiadomość 2: *Test wiadomosci, parametr1: Ania parametr2: 40*

W razie problemów podczas odwoływania się do podstawowego adresu URL (jak w przykładzie powyżej) można wykorzystać adres URL backup **https://api2.smsapi.pl/**.

3.3 Wysłanie masowe wiadomości z wykorzystaniem parametru IDX

Istnieje możliwość wysyłania masowych wiadomości z parametrem użytkownika **IDX** różnym dla każdej wiadomości. Parametr ten zwracany jest w wywołaniu zwrotnym CALLBACK z raportem doręczenia. Parametr **IDX** może mieć dowolną wartość, może on być wykorzystywany np. jako flaga użytkownika lub wewnętrzne ID z systemu klienckiego. Dodatkowo przy użyciu parametru **IDX** możliwe jest użycie parametru **check_idx** (&check_idx=1), użycie tego parametru nie pozwoli na wysłanie więcej niż jednej wiadomości z tym samym parametrem **IDX** w trakcie ostatnich 24h. W przypadku próby wysłania wiadomości z wartością parametru **IDX** która już pojawiła się dla danego użytkownika zostanie zwrócony błąd ERROR: 53. W celu przypisania różnych parametrów różnym wiadomością należy oddzielić je znakiem “pipe” co przedstawia poniższy przykład:

idx=idx1|idx2|idx3|idx4

Liczba parametrów **IDX** musi być identyczna z liczbą wysyłanych wiadomości.



3.4 Wysłanie wiadomości z wykorzystaniem Kodów rabatowych

SMSAPI umożliwia wysyłanie wiadomości z kodami rabatowymi. Aby wysłać takie wiadomości trzeba najpierw przygotować listę kodów rabatowych w panelu klienta w menu „**Funkcje dodatkowe**” → „**Kody rabatowe**”. Kody rabatowe mogą być wczytane z uprzednio przygotowanego pliku csv lub skorzystać z wbudowanego generatora kodów.

Podczas wysyłki system pobiera kody z podanej listy jednocześnie oznaczając je jako wykorzystane (jeden kod może być wykorzystany tylko raz). Aby móc wykorzystać grupę kodów do wysyłki musi być ona aktywna (data ważności nie minęła) oraz zawierać odpowiednią liczbę kodów.

Przykład:

**[https://api.smsapi.pl/sms.do?username=uzytownik&password=hasloMD5&from=nazwa
&to=48500500500,48501501501,48502502502&discount_group=kody&message=test
wiadomosci_z_kodem_rabatowym: \[%kod%\] koniec](https://api.smsapi.pl/sms.do?username=uzytownik&password=hasloMD5&from=nazwa&to=48500500500,48501501501,48502502502&discount_group=kody&message=test wiadomosci_z_kodem_rabatowym: [%kod%] koniec)**

W razie problemów podczas odwoływania się do podstawowego adresu URL (jak w przykładzie powyżej) można wykorzystać adres URL backup **<https://api2.smsapi.pl/>**.

3.5 Wysłanie wiadomości z wykorzystaniem idz.do

W celu wysłania wiadomości zawierającej skrócony link, SMSAPI udostępnia „skracacz” **idz.do**. Linki wygenerowane przy pomocy skracacza mają format **<http://idz.do/ABCD>**. Aby w treści wiadomości umieścić skrócony link należy w parametrze message dodać **`[%idzdo:adres_url%]`**. Każdy kolejny numer w wysyłce otrzyma unikalny link na podstawie którego możliwe jest śledzenie kliknięć oraz informacji m.in. o czasie wywołania linku, typie urządzenia, systemie operacyjnym.

Przykład odwołania:

**[https://api.smsapi.pl/sms.do?
username=uzytownik&password=hasloMD5&from=nazwa
&to=48500500500,48501501501,48502502502&message=test
wiadomosci_ze_skracaczem: \[%idzdo:www.smsapi.pl%\] koniec](https://api.smsapi.pl/sms.do?username=uzytownik&password=hasloMD5&from=nazwa&to=48500500500,48501501501,48502502502&message=test wiadomosci_ze_skracaczem: [%idzdo:www.smsapi.pl%] koniec)**

W razie problemów podczas odwoływania się do podstawowego adresu URL (jak w przykładzie powyżej) można wykorzystać adres URL backup **<https://api2.smsapi.pl/>**.



4. Wysyłanie wiadomości z wykorzystaniem szablonów

Przy wykorzystaniu szablonów w bardzo prosty sposób zmienić można treść SMS-ów powiadamiających (w sklepach, serwisach internetowych, klinikach itp.) bez ingerencji w kod skryptu odpowiedzialnego za wysyłanie wiadomości SMS.

W celu skorzystania z szablonów należy:

- Po zalogowaniu na stronie <http://www.smsapi.pl> dodać szablon w menu **Wiadomości SMS** → **Szablony**.
- W miejsce, w którym ma występować parametr należy wpisać [%N%] gdzie N to liczba od 1 do 4
- W celu wykorzystania szablonu w interfejsie API należy wybrać jeden z szablonów poprzez wpisanie parametru &template=nazwa_szablonu
- Oprócz podstawowych parametrów w trakcie używania szablonów dostępne są parametry:

Parametr	Opis
<i>template</i>	Nazwa szablonu
<i>paramN</i>	Parametr wstawiany w miejsce [%N%] w szablonie gdzie N to liczba od 1 do 4
<i>single</i>	Jeżeli wiadomość będzie składała się z więcej niż 160 znaków nie zostanie wysłana i zwrócony zostanie ERROR:12 (&single=1)

Przykład:

Nazwa szablonu: **Powiadom**
Treść szablonu: Witaj [%1%], Twoje zamówienie zostało zrealizowane. Numer listu przewozowego to [%2%] sledzenie przesyłek na stronie.

<https://api.smsapi.pl/sms.do?username=login&password=hasloMD5&from=nazwa&to=48600111222&template=powiadom¶m1=Marcin¶m2=BG12344423>

Treść wiadomości: Witaj **Marcin**, Twoje zamówienie zostało zrealizowane. Numer listu przewozowego to **BG12344423** sledzenie przesyłek na stronie.

W razie problemów podczas odwoływania się do podstawowego adresu URL (jak w przykładzie powyżej) można wykorzystać adres URL backup **<https://api2.smsapi.pl/>**.



5. mail2SMS – Wysyłanie SMS za pomocą e-mail

Aby wysłać SMS-a za pomocą maila należy wysłać maila według schematu:

ADRES: **sms.do@smsapi.pl**
TEMAT: **login@haslo_32znaki_w_md5**
TREŚĆ: **from=nadawca&to=numer&message=tresc wiadomosci**

UWAGA! Hasło należy podać w formie zaszyfrowanej algorytmem MD5!

Przykład:

ADRES: **sms.do@smsapi.pl**
TEMAT: **login@8456fkty567gb3bg37b357b3457b3457**
TREŚĆ: **from=nadawca&to=numer&message=tresc wiadomosci**

Dodanie parametru **raport=1** spowoduje odsyłanie maila z raportem (potwierdzenie wysłania wiadomości lub błąd – jest to przydatne w trakcie testowania usługi):

ADRES: **sms.do@smsapi.pl**
TEMAT: **login@haslo_32znaki_w_md5**
TREŚĆ: **from=nadawca&to=numer&raport=1&message=tresc wiadomosci**

Wiadomości mogą być wysyłane w kodowaniu plain / quotedprintable / base64.

Oprócz parametrów wymienionych w powyższych przykładach w mail2sms dostępne są wszystkie parametry wymienione w Punkcie 2. Wysyłanie pojedynczych SMS'ów.

W celu wysłania wiadomości Eco przez mail2SMS podać jako wartość parametru from ustawić Eco (from=Eco).

W celu wysłania wiadomości 2way przez mail2SMS podać jako wartość parametru from ustawić 2way (from=2way). Nazwa nadawcy (zmienna &from=) musi być aktywna.

WAŻNE! Numer telefonu nie może zawierać znaku plus "+" na początku. **Parametr message musi znajdować się na końcu treści wiadomości.**



6. Sprawdzenie ilości punktów na koncie

Stan konta można sprawdzić wykorzystując funkcję należącą do zestawu funkcji dodatkowych API. Specyfikacja opisująca wszystkie dostępne funkcje dodatkowe znajduje się w dokumencie Specyfikacja HTTPS API – funkcje dodatkowe

Poniżej przedstawiona jest funkcja dodatkowa służąca do sprawdzenia stanu środków pozostałych na koncie klienta.

Parametr	Opis
<i>username</i> *	Nazwa użytkownika lub główny adres e-mail przypisany do konta w serwisie SMSAPI
<i>password</i> *	Hasło do Twojego konta w serwisie SMSAPI zaszyfrowane w MD5
<i>credits</i> *	Należy podać wartość „1”
<i>details</i>	Dodatkowo wyświetlana jest ilość wiadomości Pro oraz Eco
<i>format</i>	Dla &format=json powoduje, że zwrotka z API wysyłana jest w formacie JSON

* - pole obowiązkowe

Zapytanie: **[https://api.smsapi.pl/user.do?username=uzytkownik&password=hasloMD5&credits=1&details=1&format=json](https://api.smsapi.pl/user.do?username=<u>uzytkownik</u>&password=<u>hasloMD5</u>&credits=<u>1</u>&details=<u>1</u>&format=<u>json</u>)**

I. Przykłady odpowiedzi z użyciem parametru **&format=json**:

a) w przypadku powodzenia:

```
{
  "points": 100.00, // ilość punktów dostępnych dla użytkownika w serwisie SMSAPI
  "proCount": "606", // ilość wiadomości Pro
  "ecoCount": "1428", // ilość wiadomości Eco
  "mmsCount": "333", // ilość wiadomości MMS
  "vmsGsmCount": "476", // ilość jednostek VMS do sieci GSM
  "vmsLandCount": "714" // ilość jednostek VMS do sieci stacjonarnych
}
```

II. Przykłady odpowiedzi bez użycia parametru **&format**:

a) w przypadku powodzenia:

Odpowiedź:

```
Points:<PKT>;<Pro>;<Eco>;<MMS>;<VMS_GSM>;<VMS_STAC>
```

<PKT>	ilość punktów dostępnych dla tego użytkownika w serwisie SMSAPI
<Pro>	ilość wiadomości Pro w serwisie SMSAPI
<Eco>	ilość wiadomości Eco w serwisie SMSAPI
<MMS>	ilość wiadomości MMS w serwisie SMSAPI
<VMS_GSM>	ilość jednostek VMS do sieci GSM w serwisie SMSAPI
<VMS_STAC>	ilość jednostek VMS do sieci stacjonarnych w serwisie SMSAPI

Przykład:

```
Points: 100.0000;606;1428;333;476;714
```

W razie problemów podczas odwoływania się do podstawowego adresu URL (jak w przykładzie powyżej) można wykorzystać adres URL backup **<https://api2.smsapi.pl/>**.



7. Wysyłanie wiadomości MMS

Wysyłanie wiadomości MMS odbywa się za pomocą odwołania do adresu <http://api.smsapi.pl/mms.do> (**Niezalecane!**) lub dla połączeń ssl: <https://api.smsapi.pl/mms.do> i podanie odpowiednich parametrów.

Poniższa tabela przedstawia parametry niezbędne do wysłania wiadomości MMS.

Parametr	Opis
<i>username</i> *	Nazwa użytkownika lub adres e-mail przypisany do konta w serwisie SMSAPI
<i>password</i> *	Hasło do Twojego konta w serwisie SMSAPI zaszyfrowane w MD5
<i>to</i> *	Numer odbiorcy wiadomości w formacie 48xxxxxxxx lub xxxxxxxx. Np. 48505602702 lub 505602702.
<i>group</i> *	Nazwa grupy kontaktów z książki telefonicznej, do których ma zostać wysłana wiadomość
<i>subject</i> *	Temat wiadomości MMS. Temat może zawierać maksymalnie 30 znaków, w przypadku przekroczenia limitu wiadomość jest odrzucana i zwracany jest błąd 26.
<i>date</i>	Data w formacie unixtime (&date=1287734110) lub ISO 8601 (&date=2012-05-10T08:40:27+00:00) kiedy wiadomość ma być wysłana
<i>date_validate</i>	Ustawienie „1” sprawdza poprawność formatu podanej daty. W przypadku wystąpienia błędnej daty zwrócony zostanie błąd ERROR:54
<i>smil</i> *	Wiadomość MMS w formacie SMIL 1.0 (przykład: Przykładowa wiadomość MMS w formacie SMIL (Wysyłka MMS))
<i>idx</i>	Opcjonalny parametr użytkownika wysyłany z wiadomością a następnie zwracany przy wywołaniu zwrotnym CALLBACK. Parametr idx może mieć maksymalnie 255 znaków dopuszczalne są cyfry 0 - 9 oraz litery a - z (wielkość liter nie jest rozróżniana). (&idx=123)
<i>check_idx</i>	Pozwala zabezpieczyć przed wysłaniem dwóch wiadomości z identyczną wartością parametru <i>idx</i> . W przypadku ustawienia parametru (&check_idx=1) system sprawdza czy wiadomość z takim idx już została przyjęta, jeśli tak zwracany jest błąd 53.
<i>notify_url</i>	Parametr do podawania URL do skryptu callback do odbierania raportów doręczeń. Jeżeli na koncie nie ma ustawionego stałego adresu callback dla lub jeżeli dla tej wiadomości adres ten ma być inny niż stały można go podać w parametrze ¬ify_url.
<i>test</i>	Wiadomość nie jest wysyłana, wyświetlana jest jedynie odpowiedź (w celach testowych). (&test=1)
<i>format</i>	Dla &format=json powoduje, że zwrotka z API wysyłana jest w formacie JSON

* - pole obowiązkowe

UWAGA! Parametry *group* oraz *to* są zamienne, w odwołaniu musi się pojawić jeden z tych dwóch parametrów. Brak jednego z tych dwóch parametrów lub wystąpienie ich obu spowoduje niewysłanie wiadomości oraz zwrócenie błędu ERROR: 13.

Zapytanie: [https://api.smsapi.pl/mms.do?username=uzytownik
&password=hasloMD5&to=48500500500&subject=Testowy MMS&smil=\[smil\]&format=json](https://api.smsapi.pl/mms.do?username=uzytownik&password=hasloMD5&to=48500500500&subject=Testowy MMS&smil=[smil]&format=json)



I. Przykłady odpowiedzi z użyciem parametru **&format=json**:

a) w przypadku powodzenia:

```
{
  "count": 1,
  "list": [
    {
      "id": "1463466673180734590",
      "points": "0.3000",
      "number": "48100200300",
      "date_sent": 1463466674,
      "submitted_number": "100200300",
      "status": "QUEUE"
    }
  ]
}
```

b) w przypadku błędu:

```
{
  "error": 22, // kod błędu
  "message": "Invalid smil! <smil>...</smil>" // opis błędu + załączony kod smil
}
```

II. Przykłady odpowiedzi bez użycia parametru **&format**:

a) w przypadku powodzenia:

```
OK:<id>:<points>
```

b) w przypadku błędu:

```
ERROR:<err>
```

<id>	numer ID wysłanego MMSa. Numer należy zachować w celu sprawdzenia statusu wiadomości. Parametr ten może być zmiennej długości, nieprzekraczającej 32 znaki.
<points>	ilość pobranych punktów
<err>	kod błędu (zobacz dodatek – kody błędów)

W razie problemów podczas odwoływania się do podstawowego adresu URL (jak w przykładzie powyżej) można wykorzystać adres URL backup **<https://api2.smsapi.pl/>**.



8. Wysyłanie wiadomości VMS

Wysyłanie wiadomości VMS odbywa się za pomocą odwołania do adresu <http://api.smsapi.pl/vms.do> (**Niezalecane!**) dla połączeń ssl: <https://api.smsapi.pl/vms.do> i podanie odpowiednich parametrów.

Poniższa tabela przedstawia parametry niezbędne do wysłania wiadomości VMS:

Parametr	Opis
<i>username</i> *	Nazwa użytkownika lub adres e-mail przypisany do konta w serwisie SMSAPI
<i>password</i> *	Hasło do Twojego konta w serwisie SMSAPI zaszyfrowane w MD5
<i>from</i> *	Numer dzwoniący. Pozostawienie pola pustego powoduje wysłanie wiadomości z domyślnym numerem dzwoniącym. Przyjmowane są tylko numery zweryfikowane. Numer można dodać po zalogowaniu na stronie SMSAPI, w zakładce VMS → NUMERY DZWONIĄCE.
<i>to</i> *	Numer odbiorcy wiadomości w formacie 48xxxxxxx lub xxxxxxxx. Np. 48505602702 lub 505602702.
<i>group</i> *	Nazwa grupy kontaktów z książki telefonicznej, do których ma zostać wysłana wiadomość
<i>tts</i> *	Treść wiadomości głosowej w postaci tekstu. Tekst powinien być podawany w kodowaniu UTF-8, w przeciwnym razie zostanie zwrócony błąd Error:11.
<i>file</i> *	Treść wiadomości w postaci pliku wave, akceptowalny link do pliku umieszczonego w sieci lub plik wysłany postem metodą multipart form-data
<i>date</i>	Data w formacieunixtime (&date=1287734110) lub ISO 8601 (&date=2012-05-10T08:40:27+00:00) kiedy wiadomość ma być wysłana, parametr ten jest opcjonalny, jego brak spowoduje natychmiastową wysyłkę
<i>date_validate</i>	Ustawienie „1” sprawdza poprawność formatu podanej daty. W przypadku wystąpienia błędnej daty zwrócony zostanie błąd ERROR:54
<i>try</i>	Ilość prób połączenia (dopuszczalne wartości od 1 do 6)
<i>interval</i>	Czas w sekundach po jakim powtórzone ma być połączenie w przypadku jego nieodebrania lub odrzucenia (dopuszczalne wartości od 300 do 7200s)
<i>skip_gsm</i>	Ustawienie tego parametru (&skip_gsm=1) spowoduje pominięcie telefonów komórkowych podczas wysyłki i wysłanie wiadomości tylko do numerów stacjonarnych
<i>check_idx</i>	Pozwala zabezpieczyć przed wysłaniem dwóch wiadomości z identyczną wartością parametru <i>idx</i> . W przypadku ustawienia parametru (&check_idx=1) system sprawdza czy wiadomość z takim <i>idx</i> już została przyjęta, jeśli tak zwracany jest błąd 53.
<i>notify_url</i>	Parametr do podawania URL do skryptu callback do odbierania raportów doręczeń. Jeżeli na koncie nie ma ustawionego stałego adresu callback dlr lub jeżeli dla tej wiadomości adres ten ma być inny niż stały można go podać w parametrze <i>&notify_url</i> .
<i>test</i>	Wiadomość nie jest wysyłana, wyświetlana jest jedynie odpowiedź (w celach testowych). (&test=1)
<i>format</i>	Dla &format=json powoduje, że zwrotka z API wysyłana jest w formacie JSON

* - pole obowiązkowe

UWAGA! Parametry *tts* oraz *file* są zamienne, w odwołaniu musi się pojawić jeden z tych dwóch parametrów. Brak jednego z tych dwóch parametrów lub wystąpienie ich obu spowoduje niewysłanie wiadomości oraz zwrócenie błędu ERROR: 11.

UWAGA! Parametry *group* oraz *to* są zamienne, w odwołaniu musi się pojawić jeden z tych dwóch parametrów. Brak jednego z tych dwóch parametrów lub wystąpienie ich obu spowoduje niewysłanie wiadomości oraz zwrócenie błędu ERROR: 13.



Zapytanie: **[https://api.smsapi.pl/vms.do?username=uzytkownik
&password=hasloMD5&to=48100200300&tts=Wiadomość testowa](https://api.smsapi.pl/vms.do?username=uzytkownik&password=hasloMD5&to=48100200300&tts=Wiadomość%20testowa)**

lub: **[https://api.smsapi.pl/vms.do?username=uzytkownik
&password=hasloMD5&to=48100200300&file=http://hiper.lacze.do/pliku.wav](https://api.smsapi.pl/vms.do?username=uzytkownik&password=hasloMD5&to=48100200300&file=http://hiper.lacze.do/pliku.wav)**

I. Przykłady odpowiedzi z użyciem parametru **&format=json**:

a) w przypadku powodzenia:

```
{
  "count": 1,
  "list": [
    {
      "id": "1463468935274533230",
      "points": 0.1,
      "number": "48100200300",
      "date_sent": 1463468934,
      "submitted_number": "100200300",
      "status": "QUEUE"
    }
  ]
}
```

b) w przypadku błędu:

```
{
  "error": 11, // kod błędu
  "message": "No audio file no tts!" // opis błędu
}
```

II. Przykłady odpowiedzi bez użycia parametru **&format**:

a) w przypadku powodzenia:

```
OK:<id>:<points>
```

b) w przypadku błędu:

```
ERROR:<err>
```

<id>	numer ID wysłanego MMSa. Numer należy zachować w celu sprawdzenia statusu wiadomości. Parametr ten może być zmiennej długości, nieprzekraczającej 32 znaki.
<points>	ilość pobranych punktów
<err>	kod błędu (zobacz dodatek – kody błędów)

UWAGA! Z szacunku dla abonentów wszystkich sieci, usługi głosowe dostępne są wyłącznie w godzinach od 8:00 do 22:00. Zlecenia wysyłki w pozostałych godzinach będą rozpatrywane indywidualnie. W przypadku gdy wysłanie wiadomości VMS o zadanej porze nie jest możliwe wiadomość jest przyjęta i data wysyłki zostaje ustawiona na pierwszą możliwą.



9. Odbieranie raportu doręczenia wiadomości SMS - CALLBACK

W celu sprawdzenia statusu wiadomości należy w panelu klienta z zakładce „Ustawienia API” → „Adresy Callback” ustawić adres do skryptu do którego przekazywany będzie raport doręczenia wiadomości. Aby móc wprowadzić adres callback, skrypt musi być umieszczony w podanej lokalizacji.

Np.: http://www.moja_strona.pl/status_update.php

Ważne, aby podany adres był poprawny, tzn. aby wskazany skrypt istniał na serwerze.

W momencie dodawania adresu do skryptu callback w panelu SMSAPI, weryfikowany jest on poprzez odwołanie się do niego z pustą tablicą GET.

Do skryptu wysyłane będą status wiadomości (od 1 do 5 w zależności od tego dla ilu wiadomości zmienił się status). Parametry będą podane metodą GET i będą oddzielone przecinkami np. :

```
$_GET['MsgId']=09062414383994024,09062414383994025  
$_GET['status']=403,404
```

Poniższa tabela przedstawia wysyłane parametry:

Parametr	Opis
MsgId*	ID wysłanej wiadomości. Parametr MsgId jest parametrem o zmiennej długości, ale nie większej niż 32 znaki.
status*	Kod statusu doręczenia. Lista w 'Dodatek nr 1'
status_name	Nazwa statusu doręczenia. Lista w 'Dodatek nr 1'
idx*	Opcjonalny parametr użytkownika wysłany z SMS'em
donedate*	Czas dostarczenia wiadomości w formacie unixtime
username*	Nazwa użytkownika wysyłającego wiadomość
points*	Ilość punktów pobranych za wysyłkę wiadomości
from*	Pole nadawcy z którym została wysłana wiadomość
to*	Numer na jaki wysyłana była wiadomość
mcc	Kod Mobile Country Code identyfikujący jednoznacznie kraj przynależności numeru
mnc	Kod Mobile Network Code identyfikujący jednoznacznie sieć w ramach kraju (MCC) do której numer należy.

* - należy zachować wielkość znaków!

Skrypt powinien zwracać OK, w innym przypadku nastąpią ponowne odwołania.



10. Odbieranie raportu doręczenia wiadomości MMS - CALLBACK

W celu sprawdzenia statusu wiadomości należy w panelu klienta „Ustawienia API” → „Adresy Callback” ustawić adres do skryptu do którego przekazywany będzie raport doręczenia wiadomości.

Przykład: http://www.moja_strona.pl/status_update.php

Ważne, aby podany adres był poprawny, tzn. aby wskazany skrypt istniał na serwerze.

W momencie dodawania adresu do skryptu callback w panelu SMSAPI, weryfikowany jest on poprzez odwołanie się do niego z pustą tablicą GET.

Do skryptu wysyłane będą statusy wiadomości. Parametry będą podane metodą GET np. :

```
MsgId=09062414383994024
status=404
```

Poniższa tabela przedstawia wysyłane parametry:

Parametr	Opis
MsgId*	ID wysłanej wiadomości. Parametr MsgId jest parametrem o zmiennej długości, ale nie większej niż 32 znaki.
status*	Kod doręczenia wiadomości
to*	Numer na jaki wysyłana była wiadomość
idx*	Opcjonalny parametr użytkownika wysłany z MMS'em
donedate*	Czas dostarczenia wiadomości w formacie unixtime
username*	Nazwa użytkownika z którego została wysłana wiadomość

* - należy zachować wielkość znaków!

Skrypt powinien zwracać OK (np. echo 'OK!');, w innym przypadku nastąpią ponowne odwołania.

11. Odbieranie raportu doręczenia wiadomości VMS - CALLBACK

W celu sprawdzenia statusu wiadomości należy w panelu klienta „Ustawienia API” → „Adresy Callback” ustawić adres do skryptu do którego przekazywany będzie raport doręczenia wiadomości.

Przykład: http://www.moja_strona.pl/status_update.php

Ważne, aby podany adres był poprawny, tzn. aby wskazany skrypt istniał na serwerze.

W momencie dodawania adresu do skryptu callback w panelu SMSAPI, weryfikowany jest on poprzez odwołanie się do niego z pustą tablicą GET.

Do skryptu wysyłane będą statusy wiadomości. Parametry będą podane metodą GET np. :

```
MsgId=09062414383994024
status=404
```

Poniższa tabela przedstawia wysyłane parametry:

Parametr	Opis
MsgId*	ID wysłanej wiadomości. Parametr MsgId jest parametrem o zmiennej długości, ale nie większej niż 32 znaki.
status*	Kod doręczenia wiadomości
to*	Numer na jaki wysyłana była wiadomość
idx*	Opcjonalny parametr użytkownika wysłany z VMS'em
donedate*	Czas dostarczenia wiadomości w formacie unixtime
username*	Nazwa użytkownika z którego została wysłana wiadomość
pressed*	Klawisz wybrany przez odbiorcę wiadomości podczas jej odsłuchiwania
hangup_time*	Czas trwania połączenia

* - należy zachować wielkość znaków!

Skrypt powinien zwracać OK (np. echo 'OK!');, w innym przypadku nastąpią ponowne odwołania.

12. CALLBACK dla wysyłek masowych



Do skryptu CALLBACK dla wysyłek masowych wysyłane jest odwołanie w momencie rozpoczęcia wysyłania danej wysyłki. Odwołanie zawiera parametry przedstawione w tabeli poniżej. W celu włączenia CALLBACK'a dla wysyłek masowych należy w panelu klienta „Ustawienia API” → „Adresy Callback” ustawić adres do skryptu do którego przekazywane będą statusy wysyłek masowych

Przykład: http://www.moja_strona.pl/bulk_callback.php

Ważne, aby podany adres był poprawny, tzn. aby wskazany skrypt istniał na serwerze. W momencie dodawania adresu do skryptu callback w panelu SMSAPI, weryfikowany jest on poprzez odwołanie się do niego z pustą tablicą GET.

Poniższa tabela przedstawia wysyłane parametry:

Parametr	Opis
type*	Typ wiadomości SMS/MMS/VMS
all*	Liczba wiadomości (numerów) w wysyłce.
points*	Liczba pobranych punktów za wysyłkę.
to*	Określa gdzie (za pomocą której zakładki bramki SMS) została wysłana wysyłka csv, csv_and_text lub phonebook.
info*	W przypadku wysyłki to numerów z Bazy lista grup do których zostały wysłane wiadomości.
text*	Tekst wysyłanej wiadomości.

* - należy zachować wielkość znaków!

Skrypt powinien zwracać OK (np. echo 'OK!');, w innym przypadku nastąpią ponowne odwołania.

13. CALLBACK kliknięć idz.do

W celu sprawdzenia kliknięć w link idz.do należy w panelu klienta **Ustawienia API** → **Adresy Callback** ustawić adres do skryptu do którego przekazywany będzie raport kliknięć idz.do.

Przykład: http://www.moja_strona.pl/idzdo_callback.php

Ważne, aby podany adres był poprawny, tzn. aby wskazany skrypt istniał na serwerze. W momencie dodawania adresu do skryptu callback w panelu SMSAPI, weryfikowany jest on poprzez odwołanie się do niego z pustą tablicą GET.

Do skryptu wysyłane będą parametry metodą GET np. :

MsgId = 09062414383994024
operating_system = Android

Poniższa tabela przedstawia wysyłane parametry:

Parametr	Opis
MsgId*	ID wysłanej wiadomości. Parametr MsgId jest parametrem o zmiennej długości, ale nie większej niż 32 znaki.
to*	Numer na jaki wysyłana była wiadomość
click_date*	Czas pierwszego otwarcia odnośnika w formacie unixtime
device*	Typ urządzenia
operating_system*	System operacyjny
browser*	Rodzaj przeglądarki
username*	Nazwa użytkownika z którego została wysłana wiadomość
ip*	Adres IP z którego nastąpiło kliknięcie w skrócony link
suffix*	Sufiks skróconego adresu, np. Hq9Y (dla http://idz.do/Hq9Y)

* - należy zachować wielkość znaków!

Skrypt powinien zwracać OK (np. echo 'OK!');, w innym przypadku nastąpią ponowne odwołania.



14. Odbiór SMS/MMS

Serwis **SMSAPI** oferuje również odbiór wiadomości SMS oraz MMS. Odbierać można wiadomości SMS i/lub MMS wysłane na wykupiony numer dedykowany lub odpowiedzi na wysłane wiadomości 2way.

Czas oczekiwania na wiadomość zwrotną od adresata w systemie 2Way wynosi 24h. Po tym czasie wiadomość **nie zostanie** przypisana do konta klienta.

14.1 Odbiór SMS

Odbiór wiadomości odbywać się będzie za pomocą skryptu, do którego adres należy ustawić w zakładce „Ustawienia API” → „Adresy Callback”.

Działanie skryptu opisane jest poniżej:

Po odebraniu wiadomości odwoływać będziemy się do skryptu stworzonego przez Państwa, który powinien obsługiwać następujące parametry z tablicy POST:

Parametr	Opis
sms_to	numer telefonu odbiorcy
sms_from	numer telefonu nadawcy
sms_text	treść wiadomości
sms_date	czas w postaci unixtime pobrany z SMS'a
username	nazwa użytkownika (login) do którego wiadomość została przydzielona
MsgId	id wiadomości 2way na którą jest to odpowiedź, dla odpowiedzi na numer dedykowany pole to będzie puste Parametr MsgId jest parametrem o zmiennej długości, ale nie większej niż 32 znaki.

Dane wysyłane są w kodowaniu UTF8.

Skrypt powinien zwracać OK (np. echo 'OK');, w innym przypadku nastąpią ponowne odwołania.

W momencie dodawania adresu do skryptu callback w panelu SMSAPI, weryfikowany jest on poprzez odwołanie się do niego z pustą tablicą GET.



14.2 Odbiór MMS

Odbiór wiadomości odbywać się będzie za pomocą skryptu, do którego adres należy ustawić w zakładce „Ustawienia API” → „Adresy Callback”. Działanie skryptu opisane jest poniżej:
Po odebraniu wiadomości odwoływać będziemy się do skryptu stworzonego przez Państwa, który powinien obsługiwać parametry z tablicy POST oraz FILES podane poniżej:

Tablica POST:

Parametr	Opis
mms_to	numer telefonu odbiorcy
mms_from	numer telefonu nadawcy
mms_subject	temat wiadomości
mms_date	czas w postaci unixtime pobrany z MMS'a

Tablica FILES:

Parametr	Opis
name	oryginalna nazwa wysydanego pliku
type	typ MIME wysydanego pliku (JPEG, GIF, ...)
tmp_name	tymczasowa nazwa pliku, który został wysłany na serwer
error	numer błędu (0 oznacza prawidłowe wysłanie)
size	rozmiar wysydanego pliku (w bajtach)

Dane wysyłane są w kodowaniu UTF8.

Skrypt powinien zwracać OK (np. echo 'OK!');, w innym przypadku nastąpią ponowne odwołania.

W momencie dodawania adresu do skryptu callback w panelu SMSAPI, weryfikowany jest on poprzez odwołanie się do niego z pustą tablicą GET.



15. HLR – sprawdzanie poprawności numeru

HLR (ang. Home Location Register) – Rejestr Abonentów Macierzystych – element infrastruktury telekomunikacyjnej zawierający informacje o każdym aktywnym numerze GSM. W celu skorzystania z tej opcji należy wysłać żądanie pod adres: <https://api.smsapi.pl/hlr.do> z parametrami opisanymi w tabeli poniżej. Informacje o numerach będą wysłane do skryptu do którego adres należy podać w panelu klienta na stronie <https://api.smsapi.pl> w menu „Ustawienia API” → „Adresy Callback” w polu „Adres callback - odwołania HLR”.

Ważne aby wprowadzony adres URL prowadził do istniejącego, działającego skryptu. Po sprawdzeniu numeru w systemie HLR informacja o nim zostanie przesłana do podanego adresu URL w postaci tablicy POST. W jednym odwołaniu może znajdować się do 20 numerów.

Poniższa tabela przedstawia parametry odwołania HLR:

Parametr	Opis
<i>username</i> *	Nazwa użytkownika lub adres e-mail przypisany do konta w serwisie SMSAPI
<i>password</i> *	Hasło do Twojego konta w serwisie SMSAPI zaszyfrowane w MD5
<i>number</i> *	Numer y, które mają być sprawdzone w HLR.
<i>idx</i>	Opcjonalny parametr użytkownika wysyłany z odwołaniem, a następnie zwracany przy wywołaniu zwrotnym CALLBACK. Parametr idx może mieć maksymalnie 255 znaków dopuszczalne są cyfry 0-9 oraz litery a-z (wielkość liter nie jest rozróżniana). (&idx=123)
<i>format</i>	Dla &format=json powoduje, że zwrotka z API wysyłana jest w formacie JSON.

* - pole obowiązkowe

Zapytanie: [http://api.smsapi.pl/hlr.do?
username=uzytownik&password=hasloMD5&number=48600600600,48500600700&idx=123,234&format=json](http://api.smsapi.pl/hlr.do?username=uzytownik&password=hasloMD5&number=48600600600,48500600700&idx=123,234&format=json)

I. Przykłady odpowiedzi z użyciem parametru **&format=json**:

a) w przypadku powodzenia:

```
{  
  "status": "OK",  
  "number": "48100200300",  
  "id": "46567088",  
  "price": "0.0500"  
}
```

b) w przypadku błędu:

```
{  
  "status": "ERROR",  
  "number": "481002003001", // błędny numer telefonu  
  "error": 13 // kod błędu  
}
```



II. Przykłady odpowiedzi bez użycia parametru **&format**:

a) w przypadku powodzenia:

```
OK:<number>:<id>:<points>
```

b) w przypadku błędu:

```
ERROR:<err>:<number>
```

<number>	numer przesłany do sprawdzenia w HLR
<id>	id żądania HLR w serwisie SMSAPI
<points>	ilość pobranych punktów
<err>	kod błędu (zobacz dodatek – kody błędów)

Poniższa tabela przedstawia parametry wysyłane do skryptu:

Parametr	Opis
id*	id zwrócony podczas sprawdzania numeru
number*	sprawdzany numer
mcc*	numer identyfikujący kraj (ang. Mobile Country Code)
mnc*	numer sieci w danym kraju (ang. Mobile Network Code)
info*	nazwa sieci, do której należy numer, lub opis błędu
status*	OK kiedy numer jest poprawny, FAIL kiedy numer jest błędny (np. wyłączony, nieaktywny itp.)
date*	Data w formacie UNIX timestamp, kiedy numer był sprawdzany
ported*	0 jeżeli numer jest nieprzeniesiony, 1 jeżeli numer jest przeniesiony
ported_from*	null kiedy numer jest nieprzeniesiony lub nazwa sieci z której numer został przeniesiony
idx*	Opcjonalny parametr użytkownika wysłany z zapytaniem HLR

* - należy zachować wielkość znaków!

Dane wysyłane są w kodowaniu UTF8. Lista możliwych błędów mogących pojawić się w polu **info** wraz z ich opisem znajduje się w **Dodatku nr 2 - Kody błędów**. Przykład danych przesyłanych do skryptu CALLBACK:



Przykładowe odwołanie do skryptu callback HLR:

```
Array
(
  [0] => Array
  (
    [id] => 80625
    [number] => 48600600600
    [mcc] => 260
    [mnc] => 2
    [info] => T-Mobile
    [status] => OK
    [date] => 1302703609
    [ported] => 0
    [ported_from] => null
    [belongs_to_network_since] => 1443102371
  )

  [1] => Array
  (
    [id] => 80627
    [number] => 48500600700
    [mcc] => 260
    [mnc] => 2
    [info] => ABSENT_SUBSCRIBER
    [status] => FAIL
    [date] => 1302703609
    [ported] => 0
    [ported_from] => null
    [belongs_to_network_since] => 1222263971
  )
)
```

Skrypt powinien zwracać OK (np. echo 'OK!');, w innym przypadku nastąpią ponowne odwołania.

16. Uwagi końcowe

Numery telefonu powinny być podawane w formacie 11 cyfrowym (np. 48502602702). W przypadku gdy zostanie podany numer bez prefiksu 48, zostanie on automatycznie dopisany.

Znak specjalny, to dowolny znak poza:

@£\$%&'()*+,-./0-9:;<=>?

ABCDEFGHIJKLMN OPQRSTUVWXYZäöñüšž a-zäöñüà <enter>

UWAGA! Znaki: ^ { } [] ~ \ | € <enter> liczone są **podwójnie** (w przypadku wysyłania wiadomości bez znaków specjalnych) ze względu na wymogi specyfikacji GSM.



Lista znaków diakrytycznych zamienianych na ich odpowiedniki przy ustawieniu parametru `&normalize`:

```
'normalize_chars' => array(
    'Š'=>'S', 'š'=>'s', 'Š'=>'S', 'š'=>'s', 'Đ'=>'Dj', 'đ'=>'dj', 'ž'=>'z', 'ž'=>'z', 'Ž'=>'Z', 'Ž'=>'Z', 'Ž'=>'Z',
    'ž'=>'z', 'Č'=>'C', 'č'=>'c', 'Č'=>'C', 'č'=>'c', 'Ā'=>'A', 'ā'=>'A', 'Ā'=>'A', 'ā'=>'A', 'Ā'=>'A', 'ā'=>'A', 'Ā'=>'A',
    'ā'=>'A', 'Æ'=>'A', 'Ç'=>'C', 'È'=>'E', 'É'=>'E', 'Ë'=>'E', 'ë'=>'e', 'Ê'=>'E', 'È'=>'E', 'Î'=>'I', 'Í'=>'I',
    'Ï'=>'I', 'İ'=>'I', 'Ñ'=>'N', 'Ò'=>'O', 'Ó'=>'O', 'Ô'=>'O', 'Ł'=>'L', 'ł'=>'l', 'Ń'=>'N', 'ń'=>'n', 'Ö'=>'O',
    'ö'=>'O', 'Ø'=>'O', 'Ù'=>'U', 'Ú'=>'U', 'Û'=>'U', 'Ü'=>'U', 'Ý'=>'Y', 'þ'=>'B', 'ß'=>'Ss', 'à'=>'a', 'á'=>'a',
    'â'=>'a', 'ã'=>'a', 'ä'=>'a', 'å'=>'a', 'æ'=>'a', 'ç'=>'c', 'è'=>'e', 'é'=>'e', 'ê'=>'e', 'ë'=>'e',
    'ì'=>'i', 'í'=>'i', 'î'=>'i', 'ï'=>'i', 'ð'=>'o', 'ñ'=>'n', 'ò'=>'o', 'ó'=>'o', 'ô'=>'o', 'õ'=>'o',
    'ø'=>'o', 'ù'=>'u', 'ú'=>'u', 'û'=>'u', 'ý'=>'y', 'þ'=>'b', 'ÿ'=>'y', 'Ŕ'=>'R', 'ř'=>'r',
),
```

UWAGA! Znaki spoza powyższego zestawu będą traktowane jako specjalne nawet gdy parametr `&normalize` jest ustawiony.

Tabela ilości wysłanych SMS w zależności od długość wiadomości:

Bez znaków specjalnych		Ze znakami specjalnymi (w tym PL)	
Ilość znaków	Ilość punktów	Ilość znaków	Ilość punktów
160	1 SMS	70	1 SMS
306	2 SMSy	134	2 SMSy
459	3 SMSy	201	3 SMSy
612	4 SMSy	268	4 SMSy
765	5 SMSów	335	5 SMSów
918	6 SMSów	402	6 SMSów

UWAGA!

Najnowsza specyfikacja techniczna SMSAPI znajduje się w zakładce API → Dokumentacja na stronie <http://www.smsapi.pl/>

Adresy IP z których wykonywane są odwołania callback to:

89.174.81.98, 89.174.81.101, 91.185.184.29, 91.185.185.1, 185.36.169.250 oraz 185.36.169.251



Dodatek nr 1 – Lista statusów doręczenia wiadomości

Tabela statusów doręczenia wiadomości:

Kod	Status ANG	Status	Opis
401	NOT_FOUND	Nieznaleziona	Błędny numer ID lub raport wygał
402	EXPIRED	Przedawniona	Wiadomość niedostarczona z powodu zbyt długiego czasu niedostępność numeru
403	SENT	Wysłana	Wiadomość została wysłana ale operator nie zwrócił jeszcze raportu doręczenia
404	DELIVERED	Dostarczona	Wiadomość dotarła do odbiorcy
405	UNDELIVERED	Niedostarczona	Wiadomość niedostarczona (np.: błędny numer, numer niedostępny)
406	FAILED	Nieudana	Błąd podczas wysyłki wiadomości - prosimy zgłosić
407	REJECTED	Odrzucona	Wiadomość niedostarczona (np.: błędny numer, numer niedostępny)
408	UNKNOWN	Nieznany	Brak raportu doręczenia dla wiadomości (wiadomość doręczona lub brak możliwości doręczenia)
409	QUEUE	Kolejka	Wiadomość czeka w kolejce na wysyłkę
410	ACCEPTED	Zaakceptowana	Wiadomość przyjęta przez operatora
411	RENEWAL	Ponawianie	Wykonana była próba połączenia która nie została odebrana, połączenie zostanie ponowione.
412	STOP	Zatrzymanie	Zatrzymanie

Dodatek nr 2 – Kody błędów

Tabela kodów błędów:

ERROR	Opis
8	Błąd w odwołaniu (Prosimy zgłosić)
11	Zbyt długa lub brak wiadomości lub ustawiono parametr nounicode i pojawiły się znaki specjalne w wiadomości. Dla wysyłki VMS błąd oznacz brak pliku WAV lub błąd tekstu TTS (brak tekstu lub inne niż UTF-8 kodowanie).
12	Wiadomość składa się z większej ilości części niż określono w parametrze &max_parts
13	Brak prawidłowych numerów telefonów (numer błędny, stacjonarny (w przypadku wysyłki SMS) lub znajdujący się na czarnej liście)
14	Nieprawidłowe pole nadawcy
17	Nie można wysłać FLASH ze znakami specjalnymi
18	Nieprawidłowa liczba parametrów
19	Za dużo wiadomości w jednym odwołaniu
20	Nieprawidłowa liczba parametrów IDX
21	Wiadomość MMS ma za duży rozmiar (maksymalnie 300kB)
22	Błędny format SMIL
23	Błąd pobierania pliku dla wiadomości MMS lub VMS
24	Błędny format pobieranego pliku
25	Parametry &normalize oraz &datacoding nie mogą być używane jednocześnie.
26	Za długi temat wiadomości. Temat może zawierać maksymalnie 30 znaków.



27	Parametr IDX za długi. Maksymalnie 255 znaków
28	Błędna wartość parametru <code>time_restriction</code> . Dostępne wartości to FOLLOW, IGNORE lub NEAREST_AVAILABLE.
30	Brak parametru UDH jak podany jest <code>datacoding=bin</code>
31	Błąd konwersji TTS
32	Nie można wysyłać wiadomości Eco, MMS i VMS na zagraniczne numery lub wysyłka na zagranicę wyłączona na koncie.
33	Brak poprawnych numerów
35	Błędna wartość parametru <code>tts_lector</code> . Dostępne wartości: agnieszka, ewa, jacek, jan, maja
36	Nie można wysyłać wiadomości binarnych z ustawioną stopką.
40	Brak grupy o podanej nazwie
41	Wybrana grupa jest pusta (brak kontaktów w grupie)
50	Nie można zaplanować wysyłki na więcej niż 3 miesiące w przyszłość
51	Ustawiono błędną godzinę wysyłki, wiadomość VMS mogą być wysyłane tylko pomiędzy godzinami 8 a 22 ub ustawiono kombinację parametrów <code>try</code> i <code>interval</code> powodującą możliwość próby połączenia po godzinie 22.
52	Za dużo prób wysyłki wiadomości do jednego numeru (maksymalnie 10 prób w przeciągu 60sek do jednego numeru)
53	Nieunikalny parametr <code>idx</code> . Wiadomość o podanym <code>idx</code> została wysłana w ostatnich czterech dniach lub jest zaplanowana do wysyłki w przyszłości przy wykorzystaniu parametru <code>&check_idx=1</code> .
54	Błędny format daty. Ustawiono sprawdzanie poprawności daty <code>&date_validate=1</code>
55	Brak numerów stacjonarnych w wysyłce i ustawiony parametr <code>skip_gsm</code>
56	Różnica pomiędzy datą wysyłki, a datą wygaśnięcia nie może być mniejsza niż 15 minut i większa niż 72 godzin
57	Numer znajduje się na czarnej liście dla danego użytkownika.
60	Grupa kodów o podanej nazwie nie istnieje.
61	Data ważności grupy kodów minęła.
62	Brak wolnych kodów w podanej grupie (wszystkie kody zostały już wykorzystane).
65	Brak wystarczającej liczby kodów rabatowych dla wysyłki. Liczba niewykorzystanych kodów w grupie musi być co najmniej równa liczbie numerów w wysyłce.
66	W treści wiadomości brak jest znacznika [%kod%] dla wysyłki z parametrem <code>&discount_group</code> (znacznik taki jest wymagany).
70	Błędny adres CALLBACK w parametrze <code>notify_url</code> .
72	-
74	Data wysyłki nie spełnia ograniczeń czasowych ustawionych na koncie
101	Niepoprawne lub brak danych autoryzacji. UWAGA! Hasło do API może być inne niż hasło do Panelu Klienta
102	Nieprawidłowy login lub hasło
103	Brak punktów dla tego użytkownika
104	Brak szablonu
105	Błędny adres IP (włączony filtr IP dla interfejsu API)
110	Usługa (SMS, MMS, VMS lub HLR) nie jest dostępna na danym koncie.
200	Nieudana próba wysłania wiadomości, prosimy ponowić odwołanie
201	Wewnętrzny błąd systemu (prosimy zgłosić)
202	Zbyt duża ilość jednoczesnych odwołań do serwisu, wiadomość nie została wysłana (prosimy odwołać się ponownie)



203	Zbyt wiele odwołań do serwisu. Spróbuj ponownie później.
300	Nieprawidłowa wartość pola <i>points</i> (przy użyciu pola <i>points</i> jest wymagana wartość 1)
301	Wiadomość o podanym ID nie istnieje lub jest zaplanowana do wysłania w przeciągu najbliższych 60 sekund (nie można usunąć takiej wiadomości).
400	Nieprawidłowy ID statusu wiadomości.
999	Wewnętrzny błąd systemu (prosimy zgłosić).

Tabela błędów systemu HLR:

Błąd	Opis
UNKNOWN_SUBSCRIBER	Błędny, nieaktywny numer. Błąd mający charakter stały.
ABSENT_SUBSCRIBER	Numer wyłączony lub poza zasięgiem przez dłuższy czas. Numer uznany jest jako nieaktywny (podobnie jak w przypadku UNKNOWN) jednak ma charakter tymczasowy, jeżeli numer pojawi się w zasięgu stanie się na nowo aktywny.
TELESERVICE_NOT_PROVISIONED	Numer ma zablokowaną opcję odbierania wiadomości SMS. Błąd ma charakter stały.
SYSTEM_FAILURE	Błąd systemu sieci macierzystej podczas sprawdzania numeru, ma charakter tymczasowy.
HLR_LOCAL_CANCEL / HLR_ABORT	Błąd systemu HLR dla danego numeru, ma charakter tymczasowy.
CALL_BARRED	Zablokowane połączenia przychodzące dla danego numeru. Błąd ma charakter stały.

Dodatek nr 3 – Kodowanie

Domyślnie kodowanie znaków ustawione jest na windows-1250. Jednak do wysyłania wiadomości można użyć jednego z poniżej przedstawionych rodzajów kodowania. W tym celu wykorzystać należy parametr **&encoding**.

- 'iso-8859-1'
- 'iso-8859-2'
- 'iso-8859-3'
- 'iso-8859-4'
- 'iso-8859-5'
- 'iso-8859-7'
- 'windows-1250'
- 'windows-1251'
- 'utf-8'

Przykład: [https://api.smsapi.pl/sms.do?username=uzytkownik&password=haslo&to=48500000000&encoding=utf-8&message=wiadomosc](https://api.smsapi.pl/sms.do?username=<u>uzytkownik</u>&password=<u>haslo</u>&to=<u>48500000000</u>&encoding=<u>utf-8</u>&message=<u>wiadomosc</u>)



Dodatek nr 4 – Przykładowe skrypty

Wysłanie Wiadomości SMS metodą GET

```
<?php

$params = array(
    'username' => 'uzytkownik',          //login z konta SMSAPI
    'password' => 'haslo-md5',          //lub 'password' => md5('haslo-otwarty_text'),
    'to'       => '48500000000',        //numer odbiorcy
    'from'     => 'nadawca',            //nazwa nadawcy musi być aktywna
    'message'  => 'moja wiadomosc',     //treść wiadomości
);
if ($params['username'] && $params['password'] && $params['to'] && $params['message']) {
    $data = '?'.http_build_query($params);
    $plik = fopen('https://api.smsapi.pl/sms.do'.$data,'r');
    $wynik = fread($plik,1024);
    fclose($plik);
    echo $wynik;
}

?>
```

Wysłanie Wiadomości SMS metodą POST

```
<?php

$username = 'login';                    //login z konta SMSAPI
$password = 'haslo-md5';                //lub $password = md5('haslo-otwarty_text');
$to = '5000000000';                     //numer odbiorcy
$from = 'SMSAPI';                       //nazwa nadawcy musi być aktywna
$message = "Moja wiadomosc";           //treść wiadomości
$url = 'https://api.smsapi.pl/sms.do';
$c = curl_init();
curl_setopt($c, CURLOPT_URL, $url);
curl_setopt($c, CURLOPT_POST, true);
curl_setopt($c, CURLOPT_POSTFIELDS, 'username='.$username.'&password='.$password.'&from='.$from.'&to='.$to.'&message='.$message);
curl_setopt($c, CURLOPT_RETURNTRANSFER, true);
$content = curl_exec($c);
curl_close($c);
echo $content;

?>
```

Przykład skryptu wysyłającego wiadomość MMS

```
<?php

$username = 'login';
$password = 'haslo-md5';                //lub $password = md5('haslo-otwarty_text'),
$to = '48500000000';
$subject = urlencode('SMSAPI');
$smil= urlencode('<smil><head><layout><root-layout height="600" width="425"/> <region
id="Image" top="0" left="0" height="100%" width="100%"
fit="meet"/></layout></head><body><par dur="5000ms"></img></par></body></smil>');

if($username && $password) {
    $data = '?username='.$username.'&password='.$password.'&to='.$to.'&subject='.$subject.
    $subject.
        '&smil='.$smil;
    $plik = fopen('https://api.smsapi.pl/mms.do'.$data,'r');
    $wynik = fread($plik,1024);
    fclose($plik);
    echo $wynik;
}

?>
```



Przykładowa wiadomość MMS w formacie SMIL (Wysyłka MMS)

```
<smil>
  <head>
    <layout>
      <root-layout width="160" height="135" />
      <region id="Image" width="160" height="114" left="0" top="0" /> //Rozmiar obrazka
      <region id="Text" width="160" height="20" left="0" top="120" /> //Rozmiar pola tekst
      <region id="Audio" width="106" height="1" left="0" top="120" /> //Rozmiar pola audio
    </layout>
  </head>
  <body>
    <par dur="0s">
       //Adres do obrazka
      <text src="http://adres_do_pliku_txt" region="Text" /> //Adres do pliku tekstowego
      <audio src="http://adres_do_pliku_wav" region="Audio" /> //Adres do pliku audio
    </par>
  </body>
</smil>
```

Przykładowy skrypt wysyłki VMS z wykorzystaniem biblioteki cURL

```
<?php

$path_to_file = dirname(__FILE__).'/file.wav';
$ch = curl_init('https://api.smsapi.pl/vms.do');
curl_setopt_array($ch, array(
    CURLOPT_RETURNTRANSFER => 1,
    CURLOPT_POST => 1,
    CURLOPT_POSTFIELDS => array(
        'username' => 'username',
        'password' => md5('password'),
        'to' => '48500500500',
        'file' => '@'.$path_to_file
    )
));
$response = curl_exec($ch);
echo $response;

?>
```

Przykładowy skrypt odbioru MMS

```
<?php

$IP = $_SERVER['REMOTE_ADDR'];
$received=0;

foreach($_FILES as $plik) {
    if(is_uploaded_file($plik['tmp_name'])) {
        if(move_uploaded_file($plik['tmp_name'],'mms/'.$plik['name'])) {
            $received=1;
        }
    }
}

if($received == 1) {
    echo "OK";
}

?>
```



Przykładowa wiadomość MMS w formacie SMIL z wykorzystaniem kodowania BASE64(Wysyłka MMS)

```
<smil>
  <head>
    <layout>
      <root-layout backgroundColor="#FFFFFF" height="100%" width="100%" />
      <region id="Text" top="50%" height="50%" left="0" width="100%" fit="scroll" />
    </layout>
  </head>
  <body>
    <par>
      </img>
    </par>
  </body>
</smil>
```



Odebranie raportu doręczenia (CALLBACK SMS DLR)

```
<?php

// UWAGA! Poniższy przykład jest tylko poglądowy. Nie sprawdza on danych
// wejściowych co jest zalecane dla zmiennych GET: MsgId i status, oraz nie
// sprawdza czy zmienne nie są podane przez osoby trzecie.
//
// W przypadku wysłania z wiadomością parametru 'idx' zostanie on zwrócony $_GET['idx']

if($_GET['MsgId'] && $_GET['status'] ) {
    mysql_select_db('nazwa_bazy',mysql_connect('localhost','login','haslo'));
    $arIds = explode(',',$GET['MsgId']);
    $arStatus = explode(',',$GET['status']);
    $arIdx = explode(',',$GET['idx']);

    if($arIds){
        foreach($arIds as $k => $v){
            mysql_query("UPDATE sms SET sms_status =
            '".mysql_real_escape_string($arStatus[$k])."', sms_index =
            '".mysql_real_escape_string($arIdx[$k])."' WHERE sms_id ='".mysql_real_escape_string($v)."'
            LIMIT 1");
        }

        mysql_close();
        echo "OK";
    }
}

?>
```

Przykładowa wiadomość MMS wysłana do skryptu (odbior MMS)

```
POST: Array
(
    [mms_subject] => Testowy MMS
    [mms_from] => 48600000000
    [mms_to] => 48500000000
    [mms_date_recv] => 1256031673
    [mms_date] => 1256031643
)

FILES: Array
(
    [0] => Array
        (
            [name] => 0.JPG
            [type] => image/jpeg
            [tmp_name] => /tmp/phpGXutNv
            [error] => 0
            [size] => 25097
        )

    [1] => Array
        (
            [name] => 1.txt
            [type] => text/plain
            [tmp_name] => /tmp/phpaoM6dU
            [error] => 0
            [size] => 14
        )

    [2] => Array
        (
            [name] => 3.amr
            [type] => audio/AMR
            [tmp_name] => /tmp/php6I0KEi
            [error] => 0
            [size] => 27078
        )
)
```



Historia zmian

Wersja	Data	Zmiany
Wersja 2.33	18.05.2016	1. Dodano opis zwrotek z wykorzystaniem parametru <code>&format=json</code>
Wersja 2.33	25.09.2015	1. Dodanie nowego parametru <code>belongs_to_network_since</code> dla HLR callback
Wersja 2.32	26.08.2015	1. Dodanie nowego parametru <code>allow_duplicates</code>
Wersja 2.31	04.05.2015	1. Zmiana adresów URL API
Wersja 2.30	17.02.2015	1. Dodano opis wykorzystania [%idzdo:adres_url.pl%] 2. Dodano opis CALLBACK'a dla kliknięć idz.do
Wersja 2.29.2	12.12.2014	1. Dodano nowy kod błędu 74
Wersja 2.29.1	25.02.2014	1. Zwiększono liczbę znaków dla parametru <code>idx</code>
Wersja 2.29	08.11.2013	1. Dodano parametr <code>from</code> do wysyłek wiadomości VMS
Wersja 2.28	07.10.2013	1. Dodano opis CALLBACK'a dla wysyłek masowych.
Wersja 2.27.2	04.09.2013	1. Wprowadzono nowy błąd 57.
Wersja 2.27.1	28.08.2013	1. Dodano angielskie nazwy raportów doręczeń (zwracane w bibliotekach i zwrotkach JSON).
Wersja 2.27.0	30.07.2013	1. Dodano parametr <code>idx</code> dla zapytań HLR 2. Dodano błąd 27 (za duża wartość parametru <code>idx</code>)
Wersja 2.26.0	18.07.2013	1. Dodano parametr <code>&skip_foreign</code> .
Wersja 2.25.0	23.04.2013	1. Dodano obsługę kodów rabatowych poprzez API. 2. Dodano parametr <code>&sms2way</code> dla wiadomości SMS 3. Dodano parametr <code>&notify_url</code> dla wiadomości SMS, MMS oraz VMS
Wersja 2.24.1	05.02.2013	1. Dodano opisy błędów usługi HLR. 2. Zaktualizowano listę adresów IP, z których wysyłane są odwołania CALLBACK
Wersja 2.24.0	19.10.2012	1. Dodano parametr <code>&expiration_date</code> oraz odpowiadający mu błąd 56. 2. Udostępniono możliwość zmiany lektora w wiadomościach VMS. (<code>&tts_lector</code>). 3. Nowy błąd 110 określający brak dostępu do danej usługi. 4. Dodany nowy kod błędu 36. 5. Dodano ograniczenie długości tematu dla wiadomości MMS oraz odpowiadający mu błąd 26.
Wersja 2.23.0	13.09.2012	1. Dodano parametry <code>pressed</code> oraz <code>hangup_time</code> do listy parametrów przesyłanych w odwołaniu CALLBACK z raportami doręczeń VMS. 2. Dodano parametr <code>&max_parts</code>
Wersja 2.22.0	01.08.2012	1. Dodano parametry <code>&idx</code> oraz <code>&check_idx</code> dla wysyłki wiadomości MMS oraz VMS. 2. Dodano parametr <code>&idx</code> do odwołań CALLBACK z raportami doręczeń wiadomości MMS oraz VMS. 3. Dodano parametr <code>&normalize</code> do wysyłki wiadomości SMS oraz błąd 25 związany z tym parametrem
Wersja 2.21.0	16.07.2012	1. Dodano możliwość podawania daty w formacie ISO 8601 oraz błąd 54 oznaczający błędny format daty przy zaznaczeniu <code>&date_validate=1</code> . 2. Uaktualniono sposób sprawdzania ilości środków na koncie
Wersja 2.20.0	27.03.2012	1. Dodano parametr <code>check_idx</code> do wysyłki SMS oraz błąd 53 związany z tym parametrem. 2. Dodano parametr <code>skip_gsm</code> do wysyłki VMS oraz błąd 55 związany z tym parametrem.
Wersja 2.19.0	22.03.2012	1. Opis odwołań HLR 2. Wprowadzenie nowego błędu 52. System zabezpieczający przed przypadkowym zapętleniem się skryptu.
Wersja 2.18.0	02.02.2012	1. Nowe przykłady wysyłek wiadomości
Wersja 2.17.0	13.12.2011	1. Dodane nowe kody błędów



		2. Aktualizacja opisu wysyłki VMS (usunięte efekty specjalne)
Wersja 2.16.0	31.10.2011	1. Opis parametrów <i>try</i> oraz <i>interval</i> dla wysyłek VMS
Wersja 2.15.0	14.10.2011	1. Dodano możliwość wysyłania wiadomości do grup z książki telefonicznej poprzez API 2. Dodano opis odbioru raportów doręczenia wiadomości VMS poprzez CALLBACK
Wersja 2.14.0	12.09.2011	1. Dodano opis API dla wiadomości VMS
Wersja 2.13.0	29.08.2011	1. Aktualizacja listy kodów błędów 2. Aktualizacja listy statusów wiadomości
Wersja 2.12.0	05.08.2011	1. Aktualizacja specyfikacji pod względem nowej wersji serwisu SMSAPI
Wersja 2.11.0	24.04.2011	1. Dodano adresy URL backup
Wersja 2.10.0	28.09.2010	1. Oddzielona specyfikacja SOAP od http.
Wersja 2.9.0	08.07.2010	1. Rozszerzenie funkcji sprawdzania ilości punktów w serwisie SMSAPI.pl o pokazywanie ilości dostępnych SMS.
Wersja 2.8.0	05.05.2010	1. Zmiana w zwracaniu wyniku podczas wysyłania wiadomości masowych korzystając z metody SOAP
Wersja 2.7.0	18.03.2010	1. Opis parametru fast=1
Wersja 2.6.0	24.02.2010	1. Filtr IP dla interfejsu API
Wersja 2.5.0	22.01.2010	1. Wysyłanie wiadomości MMS
Wersja 2.4.0	11.11.2009	1. Odbieranie wiadomości MMS
Wersja 2.3.0	28.08.2009	1. Opis korzystania z serwisu SMSAPI.pl za pomocą WebServices
Wersja 2.2.0	10.07.2009	1. Parametr nunicode 2. Parametr single=1 3. Sprawdzanie statusu wiadomości – kilka statusów za pomocą jednego zapytania
Wersja 2.1.0		1. Wysyłanie wiadomości Eco poprzez MailtoSMS 2. Zmiana odbioru raportów doręczenia poprzez CALLBACK
Wersja 2.0.0		1. Wysyłanie wiadomości Eco
Wersja 1.9.0		1. Wysyłanie wiadomości WAP PUSH 2. Wysyłka z wykorzystaniem parametru IDX
Wersja 1.8.0		1. Wysyłanie z wykorzystaniem szablonów 2. Wysyłka masowa ze spersonalizowaną treścią
Wersja 1.7.0		1. Wysyłka o określonej dacie/godzinie

UWAGA! Najnowsza specyfikacja techniczna SMSAPI znajduje się w zakładce **API** → **Dokumentacja** na stronie <http://www.smsapi.pl/>