

DOKUMENTACJA INTERFEJSU MYSQL

Platforma SMeSKom – instrukcja połączenia poprzez mySQL
Protokół w wersji 3.1

Autor smeskom@smeskom.pl
Data 16.06.2009
Wersja 3.1 rev.1

SOFTIKA
ROZWIĄZANIA NA MIARĘ POTRZEB



Spis treści

1 Zawartość dokumentu.....	3
2 Zmiany	3
3 Informacje ogólne.....	3
3.1 Zastosowanie.....	3
3.2 Połączenie do bazy mySQL.....	3
3.2.1 Parametry połączenia.....	4
3.2.1.1 Wersja bazy danych.....	4
3.2.1.2 Baza ustalana przy połączeniu.....	4
3.2.1.3 Kodowanie znaków.....	4
3.2.1.4 Autoryzacja.....	4
3.2.1.5 Adres i port serwera bazodanowego.....	4
3.2.1.6 Połączenie SSL.....	4
3.2.2 Ograniczenia połączenia.....	5
3.3 Tabele bazy danych.....	5
3.3.1 Tabela z SMS-ami wychodzącymi (sms_out).....	5
3.3.2 Tabela z SMS-ami przychodzącymi (sms_in).....	7
3.4 Operacje wspierane przez interfejs.....	7
3.4.1 Wysyłanie SMS-ów.....	8
3.4.2 Odbieranie SMS-ów.....	9
3.4.3 Odwołanie SMS-a zaplanowanego do wysyłki w czasie przyszłym.....	10
3.4.4 Przykłady.....	10
3.4.4.1 Wysłanie SMS-a i sprawdzenie, czy został wysłany.....	10
3.4.4.2 Wysłanie SMS-a z innym polem nadawcy niż domyślne.....	10
3.4.4.3 Optymalne sprawdzanie statusów wiadomości.....	10
3.4.4.4 Odebranie SMS-ów, które przyszły w zadanym czasie.....	11
3.4.4.5 Odwołanie zaplanowanego SMS-a.....	11

1 Zawartość dokumentu

Dokument ten opisuje interfejs mySQL pozwalający na połączenie z platformą SMeSKom.

2 Zmiany

v. 3.1

- dodano pole **sender** w tabeli sms_out, pozwalające na zmianę nadawcy SMS-a (pole ma sens tylko dla usług typu "Powiadomienia")

v. 3.0

- dodano pole **action** w tabeli sms_out, pozwalające na cofnięcie zaplanowanego SMS-a (update pola wartością 'revoke')

3 Informacje ogólne

Interfejs mySQL to możliwość odbierania i wysłania SMS-ów za pomocą bazy danych mySQL. Klient, który w swojej usłudze zaktywuje ten interfejs otrzymuje dostęp do bazy danych, która posiada tabele służące do zarządzania SMS-ami. Podłączenie do bazy może odbyć się za pomocą dowolnej aplikacji klienckiej, czy skryptu wykorzystującego interfejs bazodanowy. Baza danych mySQL jest najczęściej wykorzystywaną relacyjną bazą danych w aplikacjach webowych, dlatego podłączenie do niej nie powinno stanowić większego problemu.

3.1 Zastosowanie

Opisany w niniejszym dokumencie, interfejs mySQL ma zastosowanie jedynie w ramach platformy SMeSKom i jest jej częścią składową. Interfejs pozwala na wysyłanie i odbieranie SMS-ów w ramach wspomnianej platformy i jest to jedyne jego zastosowanie w obecnej wersji.

3.2 Połączenie do bazy mySQL

Połączenie do bazy danych powinno być utrzymywane przez cały czas komunikacji i nie należy podłączać się i rozłączać przed każdym zapytaniem mySQL. Aplikacja kliencka ma obowiązek utrzymania połączenia i w razie zerwania go (np. ze względu na zerwanie połączenia przy bezczynności, czy wystąpienie problemów z dostępem do sieci), nawiązać je na nowo.

3.2.1 Parametry połączenia

Niniejszy punkt zawiera istotne dla komunikacji parametry połączenia klienta z bazą danych.

3.2.1.1 Wersja bazy danych

Wersja bazy danych mysql : 5.0.45

3.2.1.2 Baza ustalana przy połączeniu

Wszyscy użytkownicy łączą się z serwerem ustalając bazę jako „gdbsmeskom31” (baza dla wersji 3.1 protokołu).

UWAGA: protokoły w poprzednich wersjach są i będą ciągle aktywne. Zmiana protokołu polega na zmianie bazy, z którą łączy się klient:

MySQL v3.1 – baza **gdbsmeskom31**

MySQL v3.0 – baza **gdbsmeskom3**

MySQL v2.0 – baza **gdbsmeskom2**

MySQL v1.0 – baza **gdbsmeskom**

3.2.1.3 Kodowanie znaków

Domyślne kodowanie znaków w tabelach: UTF-8.

Podłączenie się do bazy danych za pomocą innego kodowania, może spowodować przekłamanie w treści SMS-ów.

3.2.1.4 Autoryzacja

Użytkownik podłączając się do bazy używa nazwy użytkownika oraz hasła podane w Panelu administracyjnym (zakładka Interfejsy)

3.2.1.5 Adres i port serwera bazodanowego

Adres i port serwera podane są w Panelu administracyjnym (zakładka Interfejsy)

3.2.1.6 Połączenie SSL

Platforma SMeSKom pozwala na bezpieczne połączenie bazodanowe za pomocą SSL. Aplikacja kliencka musi wspierać tego typu połączenia. Nie wszystkie interfejsy na rynku wspierają połączenie SSL. Dla przykładu, aby utworzyć bezpieczne połączenie za pomocą PHP, należy użyć rozszerzenia „mysqli” (więcej pod adresem: <http://pl2.php.net/mysqli>).

Dodatkowe informacje można znaleźć w dokumentacji mysql:

<http://dev.mysql.com/doc/refman/5.0/en/secure-connections.html>

W razie pytań lub potrzeby wykorzystania bezpiecznego połączenia mysql –

prosimy o kontakt w celu otrzymania odpowiednich certyfikatów.

3.2.2 Ograniczenia połączenia

W celu uniknięcia przeciążenia serwera przez klientów bazy, użytkownicy bazodanowi są tworzeni z następującymi limitami:

MAX_QUERIES_PER_HOUR 2000 (maksymalna ilość zapytań na godzinę)

MAX_UPDATES_PER_HOUR 2000 (maksymalna ilość update'ów na godzinę)

MAX_CONNECTIONS_PER_HOUR 60 (maksymalna ilość połączeń na godzinę)

Przy tworzeniu aplikacji komunikującej się z bazą danych należy wziąć pod uwagę powyższe parametry

3.3 Tabele bazy danych

Użytkownik łącząc się z bazą danych, „widzi” w bazie do której się podłączy dwie tabele:

- sms_out
- sms_in

Wpisanie jednego wiersza do tabeli *sms_out* powoduje zainicjowanie procedury wysłania SMS-a. Odczytując z tej tabeli dane można sprawdzić, jaki jest status przeznaczonych do wysłania SMS-ów.

Odbieranie SMS-ów odbywa się poprzez tabelę *sms_in*, gdzie trafiają SMS-y, które zostały wysłane z telefonów komórkowych do usługi.

3.3.1 Tabela z SMS-ami wychodzącymi (sms_out)

Poniżej znajduje się specyfikacja tabeli *sms_out*, zawierającej SMS-y wysyłane przez użytkownika.

Kolumny insert i update mówią o rozszerzonych uprawnieniach użytkownika do działań na kolumnie tabeli (domyślnie tylko *select*).

Kolumna	Typ	Insert	Update	Opis
id	integer			wewnętrzny, unikalny numer SMS-a nadany po włożeniu wiersza
user_id	integer	X	X	identyfikator numeryczny nadawany przez użytkownika (opcjonalnie - nie ma obowiązku wypełniać tego pola)
msisdn	varchar	X		numer msisdn odbiorcy SMS-a. Więcej na temat tego pola w punkcie 3.4.1
body	text	X		treść wysyłanego SMS-a. Więcej na temat tego pola w punkcie 3.4.1
sms_type	varchar	X		typ SMS-a, „n”-normalny, „f”-wiadomość typu flash
send_after	datetime	X		Czas wysyłki SMS-a. Ustawienie tego czasu anuluje dla tego czasu limity wysyłki ustawiane w panelu.
sender	varchar	X		Zmienia pole nadawcy dla SMS-ów w usłudze typu "Powiadomienia". Pominięcie pola spowoduje użycie domyślnego pola nadawcy. Uwaga! Nowe pola nadawcy muszą zostać zgłoszone/dodane w panelu administracyjnym. Użycie pola, które nie zostało wcześniej dodane lub nie zostało zaakceptowane przez serwis, spowoduje użycie domyślnego pola.
expire_at	datetime	X		Czas po przekroczeniu, którego system nie będzie próbował wysłać SMS-a. Pominięcie tego pola spowoduje ustawienie czasu na 3 dni od daty wstawienia wiersza.
inserted_at	datetime			Czas wstawienia wiersza do tabeli
sent_at	datetime			Czas wysłania SMS-a. Niewysłany SMS będzie miał wartość NULL w tym polu.
delivered_at	datetime			Czas doręczenia SMS-a odbiorcy. SMS nieodebrany będzie miał wartość NULL w tym polu.
status	varchar			Aktualny status wysyłki SMS-a. Więcej na temat tego pola w punkcie 3.4.1
fail_code	integer			Kod błędu operacji. Lista kodów znajduje się w załączniku do niniejszego dokumentu.
action	varchar		X	Zlecenie akcji na SMS-ie - patrz punkt 3.4.3
updated_at	datetime			czas ostatniej zmiany statusu SMS-a - przydatne przy odczytywaniu statusów SMS-ów

Uwaga: Z tabeli *sms_out* nie można usunąć raz wpisanego wiersza.

3.3.2 Tabela z SMS-ami przychodzącymi (*sms_in*)

Poniżej znajduje się lista kolumn widocznych w tabeli *sms_in*, zawierająca SMS-y przychodzące.

Kolumna *update* mówi o rozszerzeniu uprawnień użytkownika do działań na kolumnie tabeli (domyślnie tylko *select*).

Kolumna	Typ	Update	Opis
id	integer		wewnętrzny, unikalny numer SMS-a nadany po włożeniu wiersza do tabeli
user_id	integer	X	identyfikator numeryczny nadawany przez użytkownika (opcjonalnie - nie ma obowiązku wypełniać tego pola)
msisdn	varchar		numer msisdn nadawcy SMS-a. Więcej na temat tego pola w punkcie 3.4.2
body	text		treść odebranego SMS-a. Więcej na temat tego pola w punkcie 3.4.2
inserted_at	datetime		Czas wstawienia wiersza do tabeli
received_at	datetime		Czas odebrania SMS-a przez platformę.

3.4 Operacje wspierane przez interfejs

Interfejs MySQL wspiera dwie podstawowe operacje:

- wysyłanie SMS-ów - za pomocą wpisania wiersza do tabeli *sms_out*
- odbieranie SMS-ów - za pomocą odczytów tabeli *sms_in*

Pozwala również na operacje dodatkowe:

- sprawdzenie statusów wysyłanych SMS-ów - za pomocą odczytu tabeli *sms_out*
- nadanie własnego identyfikatora dla odebranego SMS-a - za pomocą *update*'u tabeli *sms_out* (na polu *user_id*)
- nadanie własnego identyfikatora dla wysłanego SMS-a - przy operacji *insert*'u lub przy *update*'ie na tabeli *sms_out* (na polu *user_id*)
- odwołanie SMS-a zaplanowanego do wysyłki w czasie przyszłym

3.4.1 Wysyłanie SMS-ów

Operacja wysłania SMS-a odbywa się za pomocą wstawienia wiersza do tabeli *sms_out*. Przy tej operacji użytkownik ma do dyspozycji następujące pola:

- **msisdn** (pole obowiązkowe) - numer MSISDN. Platforma SmeSKom powinna zaakceptować większość możliwych formatów o ile numer jest napisany jednoznacznie. Przykładowe formaty, które zostaną zaakceptowane:

- +48601234567 - format zalecany
- +48 601-234-567
- 601 234 567
- (0) 601234567

UWAGA: dodatkowo dla celów testowych /integracyjnych istnieją 3 specjalne numery:

- 10001 - wysłanie zakończone statusem „wysłany” („sent”)
- 10002 - wysłanie zakończone statusem „dostarczony” („delivered”)
- 10009 - wysłanie zakończone statusem „błąd” („failed”)

wysłanie SMS-a na powyższe numery nie powoduje rzeczywistej wysyłki, a opłaty nie są pobierane

- **body** (pole obowiązkowe) - treść SMS-a. Maksymalna liczba znaków jaką można przesłać jest równa 160 (dla wiadomości jednosmsowych) lub 459 (dla wiadomości wieloczęściowych - wysłane zostaną 3 SMS-y) . Jeśli w tekście znajdują się polskie znaki i chcecie Państwo je przesłać, nie wolno przekroczyć 70 znaków (201 - dla wieloczęściowych). Jeśli długość tekstu przekracza te wartości, polskie znaki zostaną automatycznie zamienione na ich odpowiedniki bez ogonków. Więcej na ten temat dowiedzieć się można na naszych stronach („Reguły wysyłania SMS-ów”).
- **expire_at** (pole opcjonalne) - data wygaśnięcia ważności. Platforma nie pozwoli na wysłanie SMS po tym terminie. Jeśli parametr nie zostanie podany, czas wygaśnięcia zostanie ustawiony na 3 dni do przodu.
- **send_after** (pole opcjonalne) - data i czas wysłania SMS-a. Niewpisanie do tego pola żadnej wartości lub wpisanie wartości null powoduje próbę wysłania natychmiast (o ile ustawienia limitu wysyłki w panelu nie stanowią inaczej). Wpisanie daty anuluje limity i SMS zostanie wysłany o podanym czasie niezależnie od ustawień limitów.
- **sender** (pole opcjonalne) - Zmienia pole nadawcy dla SMS-ów w usłudze typu "Powiadomienia". Pominięcie pola spowoduje użycie domyślnego pola nadawcy.

Uwaga! Nowe pola nadawcy muszą zostać zgłoszone/dodane w panelu administracyjnym. Użycie pola, które nie zostało wcześniej dodane lub nie zostało zaakceptowane przez serwis, spowoduje użycie domyślnego pola.

- **sms_type** (pole opcjonalne) - typ wysyłanego SMS-a. Możliwe

wartości:

- n - wartość domyślna - zwykły SMS
- f - wyskakujący SMS (wiadomość flash)
- **user_id** (pole opcjonalne) - użytkownik może ustawić to pole wartością numeryczną i wykorzystywać je w dowolnym celu.

Dodatkowe pola, które można jedynie odczytać z bazy:

- **id** - wewnętrzny identyfikator SMS-a wychodzącego
- **status** - status wiadomości. Wartość elementu może być jedna z:
 - *new* - nowy SMS, status new zostaje ustawiony przy wpisaniu wiersza do bazy
 - *queued, processed* - system rozpoczął przetwarzanie wiadomości, wiadomość została przeznaczona do wysyłki
 - *sent* - wiadomość została poprawnie wysłana przez system
 - *delivered* - wiadomość została odebrana przez adresata (przyszedł tzw. „raport doręczenia”)
 - *failed* - nie udało się wysłać wiadomości. Informacja o błędzie została zawarta w polu *fail_code*
- **inserted_at** - czas wstawienia wiersza do bazy
- **sent_at** - czas wysłania SMS-a z systemu
- **delivered_at** - czas doręczenia SMS-a do odbiorcy.
- **fail_code** - wartość numeryczna kodu błędu związanego z wysyłanym SMS-em. Wartość 0 oznacza brak błędu i poprawne zakończenie wysyłki.
- **updated_at** - data i czas ostatniej modyfikacji statusu SMS-a przez platformę. Informacja ta jest przydatna przy sprawdzaniu statusów SMS-ów.

3.4.2 Odbieranie SMS-ów

Wszystkie odebrane przez platformę SMS-y (SMS-y które zostały wysłane z telefonów komórkowych na numer usługi) można odebrać za pomocą tabeli *sms_in*. Jeśli interfejs mySQL został utworzony jako bramka dzielona, SMS-y wysyłane do usługi muszą być poprzedzone prefiksem. W tabeli *sms_in* pole *body* zawiera tekst wiadomości już bez prefiksów.

Pola widoczne w tabeli:

- **id** - wewnętrzny, unikalny identyfikator numeryczny odebranego przez platformę SMS-a.
- **user_id** - jedyne pole, które można modyfikować w tabeli. Można tu wstawić dowolną wartość numeryczną. Głównym zastosowaniem tego pola może być powiązanie tego SMS-a z inną bazą danych lub oznaczenie go jako przeczytanego przez aplikację kliencką.
- **msisdn** - numer MSISDN telefonu, który wysłał wiadomość do usługi. Numer ten zapisany jest w formacie: „+48601234567”

- **body** - treść otrzymanej wiadomości
- **inserted_at** - czas wstawienia wiersza do bazy przez system
- **received_at** - czas odebrania SMS-a przez platformę

3.4.3 Odwołanie SMS-a zaplanowanego do wysyłki w czasie przyszłym

Użycie kolumny *send_after* pozwala na zaplanowanie wysyłki SMS-a w konkretnym czasie w przyszłości. Może się jednak zdarzyć, że chcemy anulować wysyłkę SMS-a. Do anulowania zaplanowanego SMS-a będzie potrzebny update kolumny *action*.

Wpisanie do kolumny wartości **revoke** spowoduje uruchomienie procedury odwołania SMS-a. Jeśli SMS nie został jeszcze przekazany do wysyłki (ma status „new” lub „queued” i czeka na serwerze platformy SmeSKom) nastąpi zwrot środków i wycofanie SMS-a. Po kilku sekundach w polu *status* pojawi się wartość „failed”, a kolumna *fail_code* uzyska odpowiednią wartość. Jeśli odwołanie nie powiedzie się, kolumna *status* nie zmieni swojej wartości, natomiast pole *fail_code* otrzyma odpowiedni kod błędu. Po zakończeniu procedury odwołania kolumna *action* zmieni swoją wartość na *null*.

3.4.4 Przykłady

Punkt zawiera praktyczne przykłady wykorzystania interfejsu

3.4.4.1 Wysłanie SMS-a i sprawdzenie, czy został wysłany

Wysłanie SMS-a.

```
insert into sms_out (msisdn,body,) values  
('+48601234567','Przykładowa treść SMS-a')
```

Pobranie id ostatnio wpisanego wiersza:

```
select last_insert_id()
```

Sprawdzenie statusu wysłania pojedynczego SMS-a:

```
select status,sent_at,delivered_at from sms_out where id=123  
select * from sms_out where id=123
```

3.4.4.2 Wysłanie SMS-a z innym polem nadawcy niż domyślne

Wysłanie SMS-a.

```
insert into sms_out (msisdn,body,sender) values  
('+48601234567','Przykładowa treść SMS-a','SMeSKom')
```

3.4.4.3 Optymalne sprawdzanie statusów wiadomości

Zalecana metoda odczytywania/synchronizacji statusów z własnym systemem to wykorzystanie pola *updated_at* do sprawdzania SMS-ów, których status zmienił się od czasu ostatniego odpytywania o zmiany w statusach. Tego typu odpytanie powinno wykonywać się z częstotliwością zależną od czasu

ostatniej zmiany statusu. Jeśli SMS nie został doręczony do telefonu od razu, SMSC operatora będzie próbować dostarczyć go jeszcze kilka razy, ale za każdym razem z coraz większą przerwą. W związku z tym najbardziej optymalne będzie, gdy zaraz po wysłaniu SMS-a odpytujemy się częściej, ale gdy ostatnio wysłany SMS nie został dostarczony od razu odpytujemy się rzadziej, aż do maksymalnego czasu odpytywania (np. 15 minut).

Oczekiwanie na następne odczytanie statusów powinno zwiększać się po każdorazowym odpytaniu (aż do wartości maksymalnej), a wysyłka nowego SMS-a powinna resetować ten czas do wartości minimalnej.

3.4.4.4 Odebranie SMS-ów, które przysły w zadanym czasie

Odebranie SMS-ów, które przysły w przeciągu ostatnich 30-u minut

```
select * from sms_in where inserted_at >=
timestampadd(minute,-20,now())
```

Odebranie SMS-ów, które przysły między 13-ą, a 14-ą wskazanego dnia:

```
select * from sms_in
where inserted_at between '2007-11-14 13:00:00' and
'2007-11-14 14:00:00'
```

3.4.4.5 Odwołanie zaplanowanego SMS-a

Odwołanie SMS-a o id: 1632032

```
update sms_out set action='revoke' where id=1632032
```

sprawdzenie po kilku sekundach, czy odwołanie powiodło się

```
select action,status,fail_code from sms_out where id=1632032
```

jeśli pole action zawiera null - oznacza to, że procedura odwołania została zakończona

pole status - powinno mieć wartość „failed” - SMS nie zotał wysłany

pole „fail_code” wartość 410 - poprawnie zakończona procedura odwołania

Załącznik 1. Kody błędów

Kody błędów zostały przeniesione do oddzielnego dokumentu dostępnego w dziale pliki.