

# CashBill

Specyfikacja API 1.0

**API REST**

Specyfikacja kontroli Konta systemu CashBill z wykorzystaniem API opartego na REST

## Spis treści

<b>1</b>	Jak korzystać z CashBill REST API?	3
1.1	Autoryzacja	3
1.2	Dostępne metody	3
1.3	Szczegóły protokołu	3
1.4	Opcje serwera REST	4
<b>2</b>	Historia zmian	5

# 1 Jak korzystać z CashBill REST API?

## 1.1 Autoryzacja

Użytkownik systemu CashBill może wygenerować dowolną liczbę kluczy API umożliwiających korzystanie z API REST. Każdy klucz ma przypisane niezależne od innych uprawnienia.

Zalecamy, by każda aplikacja korzystająca z REST API wykorzystywała własny klucz API. Takie rozwiązanie umożliwia ścisłą kontrolę nad dostępem do konta CashBill.

Ze względu na bezstanową naturę REST każde polecenie wydawane przez API wymaga podania klucza API. Klucz podawany jest w sekcji Query String adresu URL, np.:

```
https://cluster.cashbill.pl/api.php/test/ping/test-ping-msg?<klucz_API>
```

## 1.2 Dostępne metody

Lista dostępnych metod dla danego klucza API, wraz z opisem parametrów i szczegółową dokumentacją znajduje się w Koncie, w sekcji Klucze API.

## 1.3 Szczegóły protokołu

### Kodowanie znaków

Wszystkie parametry, wyniki i komunikaty korzystają z kodowania **utf-8**.

### Typy danych

Natura komunikacji HTTP wymaga przesyłanie parametrów w postaci ciągów znaków. Gdy w dokumentacji metody określono inny typ parametru, należy go rozumieć jako:

typ	znaczenie
string	dowolny ciąg znaków
string(xx)	dowolny ciąg znaków, o maksymalnej długości xx
int	ciąg cyfr 0-9 interpretowany jako liczba całkowita

typ	znaczenie
float	ciąg cyfr 0-9 oraz jednego znaku „.” (kropki), interpretowany jako liczba zmiennoprzecinkowa
boolean	ciągi znaków „true” lub „false”

## Wyniki operacji

Wszystkie wyniki operacji przekazywane są w formacie JSON<sup>1</sup>. Typy danych wyjściowych odpowiadają typom definiowanym przez specyfikację JSON. Poprawne wykonanie operacji jest dodatkowo sygnalizowane statusem HTTP 200/OK.

## Komunikaty błędów

W przypadku wystąpienia błędu serwer REST odpowiada statusem HTTP innym niż 200/OK. Dodatkowo, w treści odpowiedzi zawarty jest szczegółowy komunikat błędu, np.:

```
{"error":{"message":"Not authorized to perform operation"}}
```

## Zapewnienie niepowtarzalności operacji

Część zleceń wydawanych systemowi poprzez API wymaga podania unikalnego identyfikatora (tzw. **nonce**). W przypadku, gdy aplikacja korzystająca z API nie jest pewna wykonania operacji (np. wystąpił błąd w połączeniu sieciowym) może ona ponownie przesać to samo zlecenie. Jeżeli zostało one już przetworzone przez API zostanie wygenerowany stosowny komunikat błędu, bez ryzyka ponownego wykonania tej samej operacji.

Stosowanie unikalnego identyfikatora zapewnia, że polecenia zostaną wykonane tylko raz, nawet w przypadku, gdy nastąpi retransmisja danych. Identyfikator jest ważny przez godzinę i powinien zostać wygenerowany przez aplikację korzystającą z API.

### 1.4 Opcje serwera REST

Dodatkowe opcje kontrolujące serwer REST należy podać w sekcji Query String żądania URL, po tokenie i znaku '&'. W przypadku stosowania więcej niż jednej opcji należy oddzielić je znakiem przecinka.

<sup>1</sup> JavaScript Object Notation ( <http://json.org/> )

Opcja	Znaczenie
<b>test</b>	Tryb testowy. Żądania w trybie testowym są badane pod względem składni tak, jak zwykle wywołania operacji, jednak nie są przeprowadzane żadne zmiany w systemie. Odpowiedzią na żądanie testowe jest zawsze boolean true.
<b>envelope</b>	Część starszych klientów REST nie obsługuje poprawnie wyników operacji w postaci skalarów. Wykorzystanie opcji „envelope” obudowuje odpowiedzi skalarne obiektem - „kopertą”. Odpowiedź jest przekazywana w polu „return”.

Przykładowe żądanie z wykorzystaniem opcji serwera:

```
https://cluster.cashbill.pl/api.php/test/ping/test-ping-msg?<klucz_API>&test,envelope
```

## 2 Historia zmian

data	zmiany
16.06.2014	Pierwsza wersja dokumentu