

eDokumenty Api

System eDokumenty udostępnia API (Application Programming Interface) dzięki któremu jesteśmy w stanie przeprowadzać podstawowe czynności bez konieczności logowania do systemu.

Usługa jest zabezpieczona rozszerzeniem WSSecurity protokołu SOAP i wymaga podania w nagłówkach wywołania XML nazwy użytkownika i hasła.

Hasło i użytkownik to specjalne dane, które należy wprowadzić do pliku config.inc pod kluczami:

```
<?php
define('EDOK_API_LOGIN', 'edok_api_user');
define('EDOK_API_PASSWORD', 'edok_api_pass');

// Dodatkowa stała która umożliwia pominięcie autentykacji (jeśli ustawiamy na FALSE to nie sprawdza danych EDOK_API_LOGIN
// domyślnie ustawiona na TRUE, można wysłać GET lub dodatkowe nagłówki WSS
// define('EDOK_API_AUTH_MODE', TRUE);
?>
```

Wartości stałych w powyższym przykładzie konfiguracji są tylko danymi prezentacyjnymi i nie powinno się ich używać na produkcyjnej bazie.

Stałe te mogą mieć dowolne wartości ważne jednak aby te same wartości podać przy wywołaniu usługi SOAP w kliencie.

Od wersji 4.0 systemu eDokumenty jest możliwość autentykacji poprzez dowolne konto użytkownika (nie musi to być EDOK_API_LOGIN, jak we wcześniejszych wersjach).

Usługa jest dostępna pod adresem:

```
http://{host}:{port}/apps/edokumenty/classes/eDokumentyApi/EDokApiServer.php
```

Opcjonalnie od wersji 0.9.8 usługa jest dostępna pod skróconym adresem:

```
http://{host}:{port}/eDokumentyApi.php
```

Wartość {host} oraz {port} należy zamienić odpowiednimi wartościami zgodnymi z konfiguracją serwera instalacyjnego systemu eDokumenty.

Dodatkowo od wersji systemu 3.3 autentykacja nie wymaga nadpisania klienta Soap. Wystarczy do url z adresem serwisu dodać parametry GET w postaci

```
<?php
// a1 - login,
// a2 - hasło,
// a3 - symbol jednostki
/*
$sops = array(
    'location' => 'http://{host}:{port}eDokumentyApi.php?a1=' . EDOK_API_LOGIN . '&a2=' . md5(md5(EDOK_API_PASSWORD) . '_SOAP_eDok...',
    'uri' => "eDokumentyAPI",
    'encoding'=>'UTF-8'
);
?>
```

Przykład wywołania serwisu oraz funkcji dodającej kontrahenta:

```
// Plik MyService.php umieszczony w apps/edokumenty.
// MyService.php
<?php
// stałe są w config.inc i są to dane potrzebne do autentykacji usługi
require_once(' ../../framework/lib/util/Translator/Translator.inc');
require_once(' ./config.inc');
```

```

// klient SOAP systemu eDokumenty wysyłający nagłówki WSSecurity zmienić na
// odpowiednią ścieżkę w zależności od lokalizacji pliku MyService.php
require_once('./classes/eDokumentyApi/EDokApiClient.inc');

// wartość {host}:{port} zmienić na odpowiednią dla serwera
$ops = array(
    'location' => 'http://{host}:{port}eDokumentyApi.php',
    'uri' => "eDokumentyAPI",
    'encoding'=>'UTF-8'
);

// Dodatkowo od wersji 3.3 autentykacja nie wymaga nadpisania klienta Soap. Wystarczy do url z adresem serwisu dodać param
// a1 - login,
// a2 - hasło,
// a3 - symbol jednostki
/*
$ops = array(
    'location' => 'http://{host}:{port}eDokumentyApi.php?a1='.EDOK_API_LOGIN.'&a2='.md5(md5(EDOK_API_PASSWORD)).'_SOAP_eDok_'.EDOK_API_SYMBOL,
    'uri' => "eDokumentyAPI",
    'encoding'=>'UTF-8'
);

// bezpośrednie wywołanie klienta Soap
$client = new SoapClient(NULL, $ops);
*/
$client = new EDokApiClient(null, $ops);
$client->setUser(EDOK_API_LOGIN); // ten sam co w config.inc
$client->setPass(md5(EDOK_API_PASSWORD)); // to samo co w config.inc

$header = new SoapHeader('eDokumentyAPI', 'entity_symbol', DEFAULT_ENTITY_SYMBOL);
$client->__setSoapHeaders($header);

// Tworzy kontakt
$contid = NULL;

try {
    $data = array(
        'name_1' => 'SOAP TEST'.date('d H:m:s'),
        'name_2' => 'SOAPTEST',
        'nip___' => 1111111111,
        'street' => 'Główna',
        'symbol' => 'FGH99'
    );
    $contid = $client->createContact($data);
    var_dump($contid);
} catch(SoapFault $fault) {

    var_dump($fault);

    if ($fault->faultcode < 100) {
        trigger_error("SOAP Fault: (faultcode: {$fault->faultcode}, faultstring: {$fault->faultstring})", E_USER_ERROR);
    }
}

?>

```

Uwagi dotyczące haseł

1. Hasło powinno być zakodowane przy użyciu funkcji md5.

```
$password = md5('haslo');
```

Jeżeli nie korzystamy z klasy **EDokApiClient**, to dołączamy do zakodowanego hasła ciąg "_SOAP_eDok_api" i ponownie tworzymy skrót md5.

```
$password = md5(md5('haslo').'_SOAP_eDok_api');
```

Od wersji 4.0 systemu eDokumenty jest możliwość autentykacji poprzez dowolne konto użytkownika (nie musi to być *EDOK_API_LOGIN*, jak we wcześniejszych wersjach).

Uwagi dotyczące przekazywania parametrów

We wszystkich funkcjach w których parametr jest określony jako (array)\$data możliwe jest przekazanie parametrów jako ciąg json (więcej na <http://www.json.org/>). Dzięki temu nie trzeba tworzyć pseudo struktur tablic asocjacyjnych dla języków programowania, w których takich typów nie ma.

Dokumentacja API 1.8.9

Dokumentacja poszczególnych funkcji, parametrów oraz przykłady wywołań dostępne są poniżej:

```
http://{host}:{port}/apps/edokumenty/classes/eDokumentyApi/DokumentacjaAPI.txt
```

Dokumenty

- [Utwórz nowy dokument](#)
- [Dodaj pracownika do uprawnionych w dokumencie](#)
- [Dodaj grupę do uprawnionych w dokumencie](#)
- [Dodaj załącznik do dokumentu](#)
- [Ustawia meta text załącznika \(używany przy wyszukiwaniu\)](#)
- [Tworzy nowy dokument typu raport dobowy z kasy](#)
- [Pobierz dane dotyczące typu dokumentu na podstawie jednego z poniższych parametrów](#)

Produkty

- [Dodanie nowego produktu](#)
- [Aktualizacja produktu](#)
- [Usuwanie produktu](#)

Sprawy

- [Dodanie nowej sprawy](#)
- [Dodaj pracownika do uprawnionych w sprawie](#)
- [Dodaj grupę do uprawnionych w sprawie](#)
- [Pobierz dane o sprawie](#)

Kontakty

- [Dodanie nowego kontrahenta](#)
- [Pobierz dane o kontrahencie](#)
- [Wyszukanie kontrahentów w bazie](#)
- [Dodanie nowej osoby kontaktowej](#)
- [Aktualizuj dane osoby kontaktowej](#)
- [Pobierz dane wszystkich osób kontaktowych wskazanego kontrahenta](#)

Zdarzenia

- [Dodaj nowe zdarzenie](#)
- [Pobierz dane o zdarzeniu](#)
- [Zwraca listę zdarzeń według zadanych parametrów](#)

Procedury

- [Zwraca dane etapów danej procedury](#)
- [Wykonaj dany etap procedury oraz aktywuj następny](#)

Zarządzanie użytkownikami i jednostkami

- [Pobierz dane dotyczące jednostki organizacyjnej na podstawie jednego z poniższych parametrów](#)
- [Dodaje konto użytkownika systemu.](#)
- [deactivateUserAccount](#)
- [reactivateUserAccount](#)
- [Zwraca dane użytkownika systemu.](#)
- [Zwraca dane grupy użytkowników](#)
- [Dodaje użytkownika do grupy](#)
- [Usuwa użytkownika z grupy](#)
- [Dodaje jednostkę organizacyjną.](#)
- [Zwraca dane jednostki organizacyjnej.](#)
- [Przypisuje użytkownika do jednostki organizacyjnej](#)
- [Zwalnia użytkownika ze stanowiska](#)
- [setUserPrivileges](#)

Urządzenia

- [Dodaj nowe urządzenie](#)
- [Dodaj lokalizację urządzenia](#)
- [Pobierz dane urządzenia](#)
- [Pobierz lokalizację urządzenia](#)

Inne

- [Wysyłanie powiadomień](#)
- [Ustawienie cechy](#)
- [Podpisz i wyślij dokument na emaila](#)
- [Wykonuje raport i zwraca jego wynik](#)
- [Publikacja teczki RWA w danej jednostce organizacyjnej](#)

MRD

- [Pobiera definicję rejestru](#)
- [Dodanie pozycji do rejestru](#)
- [Pobiera datę ostatniej modyfikacji dla całego rejestru](#)