

Title: Dodanie nowej sprawy

Subject: eDokumenty - elektroniczny system obiegu dokumentów, workflow i CRM -
DeployerGuide/Others/eDokumentyApi/CreateProcess

Version: 16

Date: 04/19/25 10:36:01

Table of Contents

Dodanie nowej sprawy

3

Dodanie nowej sprawy

Definicja parametrów:

```
<?php

/**
 * Tworzy nową sprawę
 *
 * @param data Array - Tablica z parametrami
 * Szczegóły:
 * @param data['dscrpt'] String - opis sprawy
 * @param data['symbol'] String - symbol sprawy. Jeśli nie zostanie podany system nada go automatycznie
 * @param data['briefcase_symbol'] String - symbol teczki. Parametr zamienny z dsexid
 * @param data['dsexid'] int - identyfikator teczki (doss_extract_list:dsexid). Parametr zamienny z briefcase_symbol
 * @param data['orunid'] int - identyfikator jednostki organizacyjnej, pod którą zostanie utworzona sprawa (organization_units:orunid)
 * @param data['rspoid'] int - identyfikator stanowiska osoby odpowiedzialnej w sprawie (organization_units:orunid). Parametr zamienny z rspoid
 * @param data['rsuid'] int - identyfikator osoby odpowiedzialnej w sprawie (users:usr_id). Parametr zamienny z rspoid
 * @param data['prtpid'] int - identyfikator procedury (procedures_def:prtpid). Domyślnie sprawa jest tworzona z procedurą
 * @param data['prtype'] int - typ sprawy: 1 - terminowa, 2 - stała. Domyślnie 2 czyli stała
 * @param data['comnts'] String - uwagi
 * @param data['contact_symbol'] String - symbol kontaktu (contacts:symbol). Parametr zamienny z contid
 * @param data['contid'] int - identyfikator kontaktu (contacts:contid). Parametr zamienny z contact_symbol
 * @param data['copeid'] int - identyfikator osoby kontaktowej (contact_persons:copeid)
 * @param data['fxtrid'] int - to ilość dni na zakończenie sprawy poczynawszy od Daty wszczęcia
 * @param data['rel_to'] int - identyfikator sprawy nadrzędnej, jeżeli chcemy podpiąć sprawę jako podrzędną
 * @param data['pr_sta'] String - kod statusu właściwy dla sprawy. Jeśli przypisano statusy per teczka kod musi wskazywać
 * @param data['forepa'] Numeric - Prognozowana wartość wpływów.
 * @param data['currpa'] Numeric - Waluta dla pola forepa
 * @param data['foreda'] String - Prognozowana data sprzedaży
 * @param data['foreco'] Numeric - Prognozowany koszt zlecenia
 * @param data['currco'] Numeric - Waluta dla pola foreco
 * @param data['credib'] String - Szacunek (prawdopodobieństwo otrzymania zlecenia). Dane: 10%, 25%, 50%, 75%, 100%
 * @param data['devcid'] Int - Identyfikator urządzenia (devices:devcid) jeśli sprawa dotyczy urządzenie z serwisu
 * @param data['projid'] Int - Identyfikator projektu (processes:projid) jeśli sprawa należy do projektu
 * @param data['campid'] Int - Identyfikator kampanii (campaigns:campid) jeśli sprawa należy do projektu
 * @param data['optime'] String - Data wszczęcia w formacie YYYY-MM-DD
 * @param data['cltime'] String - Data zakończenia w formacie YYYY-MM-DD. Datę tę ustawia system podczas załatwiania sprawy
 *
 * @return Int - id sprawy jeśli sukces (processes:prc_id), 0 w razie niepowodzenia
 *
 * @throws Exception - SoapFault
 */
Int createProcess(Array data)

?>
```

Przykłady wywołań:

```
// Plik MyService.php umieszczony w apps/edokumenty.
// MyService.php
<?php

define('EDOK_API_LOGIN', 'developer');
define('EDOK_API_PASSWORD', 'developer');
define('DEFAULT_ENTITY_SYMBOL', 'demo');

require_once('./classes/eDokumentyApi/EDokApiClient.inc');
```

```

$options = array(
    'location' => 'http://{host}:{port}/eDokumentyApi.php',
    'uri' => "eDokumentyAPI",
    'encoding'=>'UTF-8'
);

$client = new EDokApiClient(NULL, $options);
$client->setUser(EDOK_API_LOGIN);
$client->setPass(md5(EDOK_API_PASSWORD));
$header = new SoapHeader('eDokumentyAPI', 'entity_symbol', DEFAULT_ENTITY_SYMBOL);
$client->__setSoapHeaders($header);

$prc_id = NULL;

try {
    $data = array(
        'dscrip' => 'SOAP TEST '.date('d H:m:s'),
        'briefcase_symbol' => 'DK.023',
        'orunid' => 49,
        'rspoid' => 54,
        'fxtrid' => 60,
        'prtpid' => 1,
    );
    $prc_id = $client->createProcess($data);
    var_dump($prc_id);
} catch(SoapFault $fault) {
    var_dump($fault);

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

?>

```