Nadanie/odebranie uprawnienia użytkownikowi

Definicja parametrów:

```
<?php

   /**
    * Nadaje/odbiera uprawnienie użytkownikowi
    *
    * @param integer usr_id - Identyfikator użytkownika
    * @param string  define - Symbol uprawnienia (wartość z kolumny right_def.define)
    * @param boolean to_set - Wartość (true - nadanie uprawnienia, false - odebranie uprawnienia
    *
    * @access public
    * @return boolean - true, jeżeli operacja zakończyła się sukcesem
    */
boolean setUserPrivilages(integer usr_id, string define, boolean to_set);

?>
```

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);

$depoid = NULL;

try {
   $usr_id = 13;
   $define = 'bswfms.mails';
   $to_set = true;

   $result = $client->setUserPrivilages($usr_id, $define, $to_set);

   var_dump($result);
} catch(SoapFault $fault) {
   var_dump($fault);

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

```
    }
}

?>
```

Powyższy kod udostępniony jest na licencji LGPL http://www.gnu.org/licenses/lgpl-3.0.txt