



## **TaskManager API** (v1.6)

TaskManager API je implementováno webovými službami, protokolem XML-RPC. Následuje výčet webových služeb:

- **createItem()**
  - funkce pro vytvoření úkolu
  - parametry
    - (string) **username** – uživatelské jméno
    - (string) **pass** – heslo uživatele
    - (array) **itemDetail** – pole obsahující podrobnosti o úkolu
      - (int) **projectId** – identifikátor projektu (viz. `getProjects`)
      - (string) **projectName** – volitelný atribut, pokud nastaven vytvoří se nový projekt
      - (int) **priority** – identifikátor priority (viz. `getPriorities`)
      - (int) **context** – identifikátor typu úkolu (viz. `getContexts`)
      - (string) **title** – název úkolu
      - (string) **description** – popis úkolu
      - (date) **deadlineDate** – termín úkolu (formát (Y-m-d))
      - (int) **expectedDuration** – odhadovaná délka práce (hod.)
      - (int) **showPrivate** – nastavení zda se jedná o privátní úkol nebo ne
        - 2 – privátní úkol
        - 1 – přístupný i ostatním
      - (int) **memberId** – identifikátor uživatele (viz. `GetMembers`), pokud není nastaven, použije se uživatel vytvářející úkol
      - (int) **authorId** – identifikátor zadavatele (viz. `GetMembers`), pokud není nastaven, použije se uživatel vytvářející úkol
      - (int) **flag** – příznak urgency, pokud nastaven odesle se SMS
      - (int) **deadlineNotificationId** – identifikátor intervalu připomínání (viz. `getDeadlineNotifications`)
      - (int) **groupId** – identifikátor skupiny úkolu (viz. `getGroups`)
- **removeItem()**
  - funkce pro odstranění úkolu
  - parametry
    - (string) **username** – uživatelské jméno
    - (string) **pass** – heslo uživatele
    - (int) **itemId** – identifikační číslo úkolu
  - návratová hodnota
    - v případě úspěchu vrátí zpět identifikátor
    - v případě neúspěchu vrátí 0
- **getProjects()**
  - funkce pro získání seznamu projektu
  - parametry
    - (string) **username** – uživatelské jméno
    - (string) **pass** – heslo uživatele
  - návratová hodnota
    - (array) v případě úspěchu vrátí seznam
    - v případě neúspěchu vrátí 0
- **getContexts()**
  - funkce pro získání seznamu typu úkolu

- parametry
  - (string) **username** – uživatelské jméno
  - (string) **pass** – heslo uživatele
- návratová hodnota
  - (array) v případě úspěchu vrátí seznam
  - v případě neúspěchu vrátí 0
- **getMembers()**
  - funkce pro získání seznamu uživatelů
  - parametry
    - (string) **username** – uživatelské jméno
    - (string) **pass** – heslo uživatele
  - návratová hodnota
    - (array) v případě úspěchu vrátí seznam
    - v případě neúspěchu vrátí 0
- **getGroups()**
  - funkce pro získání seznamu skupin úkolů
  - parametry
    - (string) **username** – uživatelské jméno
    - (string) **pass** – heslo uživatele
  - návratová hodnota
    - (array) v případě úspěchu vrátí seznam
    - v případě neúspěchu vrátí 0
- **getDeadlineNotifications()**
  - funkce pro získání seznamu intervalů připomínek
  - parametry
    - (string) **username** – uživatelské jméno
    - (string) **pass** – heslo uživatele
  - návratová hodnota
    - (array) v případě úspěchu vrátí seznam
    - v případě neúspěchu vrátí 0
- **getPriorities()**
  - funkce pro získání seznamu priorit úkolů
  - parametry
    - (string) **username** – uživatelské jméno
    - (string) **pass** – heslo uživatele
  - návratová hodnota
    - (array) v případě úspěchu vrátí seznam
    - v případě neúspěchu vrátí 0
- **getTasks()**
  - funkce pro získání seznamu položek uživatele
  - parametry
    - (string) **username** – uživatelské jméno
    - (string) **pass** – heslo uživatele
    - (int) **groupId** – identifikátor skupiny úkolů (viz. GetGroups)
    - (int) **recordFrom** – pro omezení počtu vrácených úkolů (má význam offsetu v příkazu SELECT, typická hodnota pro první sadu vrácených úkolů je 0)
    - (int) **recordTo** – pro omezení počtu vrácených úkolů shora (prosím používat číslo maximálně 100). Další informace níže u popisu návratové hodnoty a její části `totalRecords`

- [(array) **filter**] – pole s filtrem (nepovinné), pole je asociativní: klíčem je název filtru, hodnotou je vnořené pole hodnot, které chceme filtrovat; dostupné filtry jsou:
  - authorFilter – filtr zadavatelů (hodnotami mohou být identifikátory získané pomocí **getMembers**)
  - memberFilter – filtr přiřazených uživatelů (hodnotami mohou být identifikátory získané pomocí **getMembers**)
  - priorityFilter – filtr priorit (hodnotami mohou být identifikátory získané pomocí **getPriorities**)
  - contextFilter – filtr typů úkolů (hodnotami mohou být identifikátory získané pomocí **getContexts**)
  - projectFilter – filtr projektů (hodnotami mohou být identifikátory získané pomocí **getProjects**)
  - lastChangeDateFilter – filtrování úkolů dle data poslední změny (atributy filtru jsou <datum\_od>, <datum\_do>)
- navratova hodnota
  - (array) v případě úspěchu vrátí pole, obsahující následující položky:
    - jednotlivé tasky (klíčem pole je přímo ID tasku)
    - totalRecords – celkový počet úkolů vyhovujících zadanému filtru (tento počet se nutně nemusí rovnat počtu vrácených úkolů, protože počet vrácených úkolů je z výkonnostních důvodů a z důvodu ochrany proti DoS útokům limitován na max.100 záznamů).
  - v případě neúspěchu vrátí 0
- **getTask()**
  - funkce pro získání úkolu
  - parametry
    - (string) **username** – uživatelské jméno
    - (string) **pass** – heslo uživatele
    - (int) **taskId** – identifikační číslo úkolu
  - navratova hodnota
    - (array) v případě úspěchu vrátí úkol
    - v případě neúspěchu vrátí 0

## Příklady PHP skriptů pro použití výše zmíněných webových služeb

(žlutě podbarvené části je nutné změnit)

Poznámka: je nutné mít na serveru nainstalovanou PHP extension pro XML-RPC. Např. na Fedora linux takto: `yum install php-xmlrpc`

### Vytvoření úkolu

```
<?php
testCreate();

function testCreate(){
    echo "<pre>";
    //server url
    $server= "http://taskmanager.cz/company/index.php?ctrl=XmlRpc";

    //attributes for createItem
```

```

$par = array(
    "username"=>"user1",
    "pass"=>"pass1",
    "itemDetail" =>array(
        "projectId" => "1",
        "priority" => "1",
        "context" => "1",
        "title" => "test nazev",
        "description" => "test popis",
        "deadlineDate" => "2011-10-05",
        "expectedDuration" => "5",
        "showPrivate" => "1",
        "memberId" => "1",
        "authorId" => "1",
        "flag" => "1",
        "deadlineNotificationId" => "1",
        "groupId" => "1",
    ),
);

$res = xmlrpc($server, "createItem", $par);

print_r($res);
}

function xmlrpc($url, $method, $params, $encoding = 'utf-8') {
    $context = stream_context_create(array('http' => array(
        'method' => "POST",
        'header' => "Content-Type: text/xml",
        'content' => xmlrpc_encode_request($method, $params, array('encoding' =>
$encoding))
    )));
    return xmlrpc_decode(file_get_contents($url, false, $context),
$encoding);
}
?>

```

### Smazání úkolu

```

<?php
    testDelete();

    function testDelete(){
        echo "<pre>";
        //server url
        $server= "http://taskmanager.cz/company/index.php?ctrl=XmlRpc";

        //attributes
        $par = array(
            "username"=>"user1",
            "pass"=>"pass1",
            "itemDetail" => array(
                "itemId"=>"2",
            ),
        );

        $res = xmlrpc($server, "removeItem", $par);
    }
}

```

```

        print_r($res);
    }

    function xmlrpc($url, $method, $params, $encoding = 'utf-8') {
        $context = stream_context_create(array('http' => array(
            'method' => "POST",
            'header' => "Content-Type: text/xml",
            'content' => xmlrpc_encode_request($method, $params, array('encoding' =>
$encoding))
        )));
        return xmlrpc_decode(file_get_contents($url, false, $context),
$encoding);
    }
?>

```

### Získání seznamu projektů založených v systému TaskManager

```

<?php
    testProjectList();

    function testProjectList(){
        echo "<pre>";
        //server url
        $server= "http://taskmanager.cz/company/index.php?ctrl=XmlRpc";

        //attributes
        $par = array(
            "username"=>"user1",
            "pass"=>"pass1",
        );

        $res = xmlrpc($server,"getProjects",$par);

        print_r($res);
    }

    function xmlrpc($url, $method, $params, $encoding = 'utf-8') {
        $context = stream_context_create(array('http' => array(
            'method' => "POST",
            'header' => "Content-Type: text/xml",
            'content' => xmlrpc_encode_request($method, $params, array('encoding' =>
$encoding))
        )));
        return xmlrpc_decode(file_get_contents($url, false, $context),
$encoding);
    }
?>

```

### Získání seznamu úkolů v systému TaskManager s použitím filtru

*Chceme získat úkoly ze skupiny s ID 2 od zadavatelů s ID 1 nebo 3.*

```

<?php
    testTaskList();

```

```

function testTaskList(){
    echo "<pre>";
    //server url
    $server= "http://taskmanager.cz/company/index.php?ctrl=XmlRpc";

    //attributes
    $par = array(
        "username"=>"user1",
        "pass"=>"pass1",
        "groupId"=>"2",
        "filter"=> array(
            "authorFilter" => array(1, 3)
        ),
        'recordFrom' => 0,
        'recordTo' => 100
    );

    $res = xmlrpc($server, "getTasks", $par);

    print_r($res);
}

function xmlrpc($url, $method, $params, $encoding = 'utf-8') {
    $context = stream_context_create(array('http' => array(
        'method' => "POST",
        'header' => "Content-Type: text/xml",
        'content' => xmlrpc_encode_request($method, $params, array('encoding' =>
$encoding))
    )));
    return xmlrpc_decode(file_get_contents($url, false, $context),
$encoding);
}
?>

```

a podobně pro ostatní webové služby.