# REST API

## Zdroj pro příklad je zde

<https://codeofaninja.com/2017/02/create-simple-rest-api-in-php.html>

PHP knihovna cURL představuje dobrý způsob, jak je možné stahovat data z cizího serveru.

Na základě dotazu sestaví HTTP request, který odešle na cílový server a po stažení obsahuje API pro (relativně) snadnou práci s daty.

Na rozdíl od nativní funkce file\_get\_contents (přes kterou můžeme také pokládat HTTP dotazy) nabízí mnohem lepší možnost konfigurace a stránky/soubory stahuje jako reálný prohlížeč.

Funkce file\_get\_contents interně knihovnu cURL používá, jenom nemá tak podrobné možnosti konfigurace.

## Detekce cURL módu v requestu

Často se hodí detekovat, jestli byl aktuální požadavek položen prostřednictvím cUrl, nebo klasicky v prohlížeči.

V PHP pro to neexistuje přímá implementace, ale můžeme si napsat jednoduchou funkci sami:

/\*\*

\* @return bool

\* @author Jan Barášek

\*/

function isCurl(): bool

{

return \stripos($\_SERVER['HTTP\_USER\_AGENT'], 'curl') !== false;

}

Pokud disponujete Linuxem a jeho Terminálem, popřípadě jste na Macu, vyzkoušejte tento příkaz:

curl https://php.baraja.cz/curl

Příkaz provede interní request na tento web a vrátí výsledek.

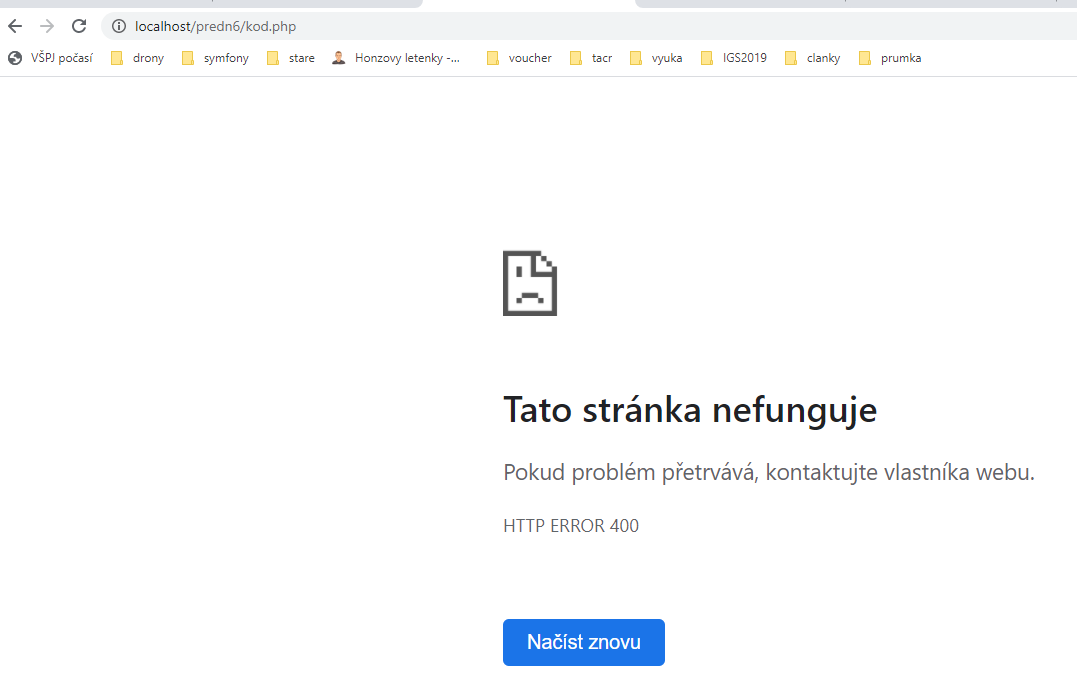
Kdyby aplikace neobsahovala detekce cURL requestu, bude vráceno HTML, jako kdyby šel request z prohlížeče. Protože se ale typy requestů detekují, nic nebrání tomu, abychom vrátili očištěný článek v Markdownu.

Výhodou jsou pak mnohem lépe očištěná data. Uživateli v prohlížeči zobrazíme naformátované HTML, zatímco robotovy jen základní obsah.

## http\_response\_code

Nastavte stavový kód odpovědi HTTP na 404:

<?php  
http\_response\_code(400);  
?>



## Kódování asociativního pole

<?php  
$age = array("Peter"=>35, "Ben"=>37, "Joe"=>43);  
  
echo json\_encode($age);  
?>

<https://www.w3schools.com/php/showphp.php?filename=demo_json_encode1>

Jak kódovat indexované pole do pole JSON:

<?php  
$cars = array("Volvo", "BMW", "Toyota");  
  
echo json\_encode($cars);  
?>

https://www.w3schools.com/php/showphp.php?filename=demo\_json\_encode2

## Funkce json\_decode () PHP

<?php  
$jsonobj = '{"Peter":35,"Ben":37,"Joe":43}';  
  
var\_dump(json\_decode($jsonobj));  
?>

<https://www.w3schools.com/php/showphp.php?filename=demo_json_decode1>

Ukládejte data JSON do proměnné PHP a poté je dekódujte do asociativního pole PHP:

<?php  
$jsonobj = '{"Peter":35,"Ben":37,"Joe":43}';  
  
var\_dump(json\_decode($jsonobj, true));  
?>

<https://www.w3schools.com/php/showphp.php?filename=demo_json_decode2>

Jak přistupovat k hodnotám z objektu PHP:

<?php  
$jsonobj = '{"Peter":35,"Ben":37,"Joe":43}';  
  
$obj = json\_decode($jsonobj);  
  
echo $obj->Peter;  
echo $obj->Ben;  
echo $obj->Joe;  
?>

<https://www.w3schools.com/php/showphp.php?filename=demo_json_decode3>

Jak přistupovat k hodnotám z asociativního pole PHP:

<?php  
$jsonobj = '{"Peter":35,"Ben":37,"Joe":43}';  
  
$arr = json\_decode($jsonobj, true);  
  
echo $arr["Peter"];  
echo $arr["Ben"];  
echo $arr["Joe"];

<https://www.w3schools.com/php/showphp.php?filename=demo_json_decode4>

# htmlspecialchars

převede speciální znaky v textu na HTML entity. Mezi speciální znaky patří: &amp;, &#039;, &quot;, &lt;, &gt;

<?php

$s = 'Vítej ve fóru, tvůj problém vyřeší tag </body>, nemáš totiž ukončené body.';

echo htmlspecialchars($s);

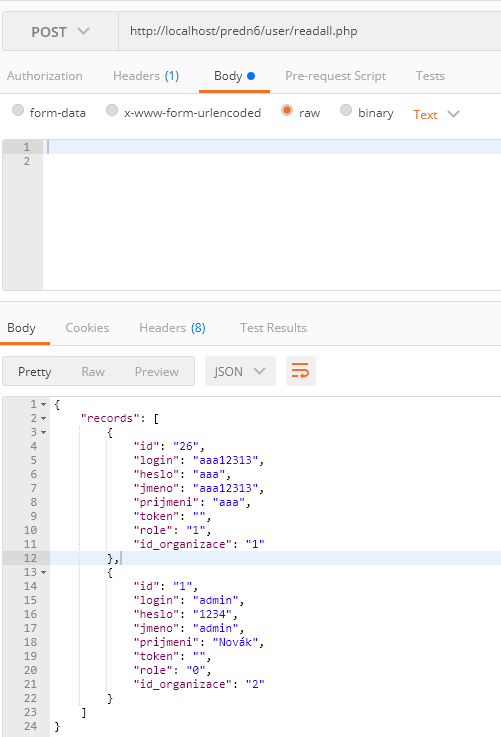
# Vysvětlení celého příkladu na REST API

## Objekt USER

<?php  
class User{  
   
 *// database connection and table name* private $conn;  
 private $table\_name = "user";  
   
 *// object properties* public $id;  
 public $login;  
 public $jmeno;  
 public $prijmeni;  
 public $heslo;  
 public $id\_organizace;  
 public $token;  
 public $role;  
   
   
 *// constructor with $db as database connection* public function \_\_construct($db){  
 $this->conn = $db;  
 }  
 *// read products*function read($keyword){  
  
 *// select all query* $query = "SELECT  
 p.id, p.login, p.jmeno, p.prijmeni , p.heslo, p.id\_organizace, p.role, p.token   
 FROM  
 " . $this->table\_name . " p  
   
 WHERE  
 p.id\_organizace=?  
 ORDER BY  
 p.prijmeni DESC";  
  
 *// prepare query statement* $stmt = $this->conn->prepare($query);  
 $stmt->bindParam(1, $keyword, PDO::*PARAM\_INT*);  
 *// execute query* $stmt->execute();  
  
 return $stmt;  
}  
 function read\_all(){  
  
 *// select all query* $query = "SELECT  
 p.id, p.login, p.jmeno, p.prijmeni , p.heslo, p.id\_organizace, p.role, p.token   
 FROM  
 " . $this->table\_name . " p  
   
 ORDER BY  
 p.id DESC";  
  
 *// prepare query statement* $stmt = $this->conn->prepare($query);  
  
 *// execute query* $stmt->execute();  
  
 return $stmt;  
 }  
function create(){  
   
 *// query to insert record* $query = "INSERT INTO  
 " . $this->table\_name . "  
 SET  
 login=:login, heslo=:heslo, jmeno=:jmeno, prijmeni=:prijmeni, token=:token, role=:role, id\_organizace=:id\_organizace";  
   
 *// prepare query* $stmt = $this->conn->prepare($query);  
   
 *// sanitize  
 //$this->id=htmlspecialchars(strip\_tags($this->id));* $this->login=*htmlspecialchars*(*strip\_tags*($this->login));  
 $this->heslo=*htmlspecialchars*(*strip\_tags*($this->heslo));  
 $this->jmeno=*htmlspecialchars*(*strip\_tags*($this->jmeno));  
 $this->prijmeni=*htmlspecialchars*(*strip\_tags*($this->prijmeni));  
 $this->token=*htmlspecialchars*(*strip\_tags*($this->token));  
 $this->role=$this->role;  
 $this->id\_organizace=$this->id\_organizace;  
  
   
 *// bind values* $stmt->bindParam(":login", $this->login);  
 $stmt->bindParam(":heslo", $this->heslo);  
 $stmt->bindParam(":jmeno", $this->jmeno);  
 $stmt->bindParam(":prijmeni", $this->prijmeni);  
 $stmt->bindParam(":token", $this->token);  
 $stmt->bindParam(":role", $this->role);  
 $stmt->bindParam(":id\_organizace", $this->id\_organizace);  
   
 *// execute query* if($stmt->execute()){  
 return true;  
 }  
   
 return false;  
   
}  
  
 function delete(){  
   
 *// delete query* $query = "DELETE FROM " . $this->table\_name . " WHERE id = ?";  
   
 *// prepare query* $stmt = $this->conn->prepare($query);  
   
 *// sanitize* $this->id=*htmlspecialchars*(*strip\_tags*($this->id));  
   
 *// bind id of record to delete* $stmt->bindParam(1, $this->id);  
   
 *// execute query* if($stmt->execute()){  
 return true;  
 }  
   
 return false;  
   
}   
  
  
*// update the product*function update(){  
   
 *// update query* $query = "UPDATE  
 " . $this->table\_name . "  
 SET  
 login = :login,  
 heslo = :heslo,  
 jmeno = :jmeno,  
 prijmeni = :prijmeni,  
 token = :token,  
 role = :role,  
 id\_organizace = :id\_organizace  
 WHERE  
 id = :id";  
   
 *// prepare query statement* $stmt = $this->conn->prepare($query);  
   
 *// sanitize* $this->id=*htmlspecialchars*(*strip\_tags*($this->id));  
 $this->jmeno=*htmlspecialchars*(*strip\_tags*($this->login));  
 $this->heslo=*htmlspecialchars*(*strip\_tags*($this->heslo));  
 $this->jmeno=*htmlspecialchars*(*strip\_tags*($this->jmeno));  
 $this->prijmeni=*htmlspecialchars*(*strip\_tags*($this->prijmeni));  
 $this->token=*htmlspecialchars*(*strip\_tags*($this->token));  
 $this->role=*htmlspecialchars*(*strip\_tags*($this->role));  
 $this->id\_organizace=*htmlspecialchars*(*strip\_tags*($this->id\_organizace));  
  
 *// bind new values* $stmt->bindParam(':id', $this->id);  
 $stmt->bindParam(":login", $this->login);  
 $stmt->bindParam(":heslo", $this->heslo);  
 $stmt->bindParam(":jmeno", $this->jmeno);  
 $stmt->bindParam(":prijmeni", $this->prijmeni);  
 $stmt->bindParam(":id\_organizace", $this->id\_organizace);  
 $stmt->bindParam(":token", $this->token);  
 $stmt->bindParam(":role", $this->role);  
  
   
 *// execute the query* if($stmt->execute()){  
 return true;  
 }  
  
 return false;  
}  
 function login($keyword){  
  
 *// select all query* $query = "SELECT  
 p.id, p.login, p.heslo, p.jmeno, p.prijmeni, p.role, p.id\_organizace   
 FROM  
 " . $this->table\_name . " p  
 WHERE  
 p.login=?";  
  
 *// prepare query statement* $stmt = $this->conn->prepare($query);  
 $stmt->bindParam(1, $keyword, PDO::*PARAM\_STR*, 100);  
  
 *// execute query* $stmt->execute();  
  
 return $stmt;  
 }  
  
  
  
  
}

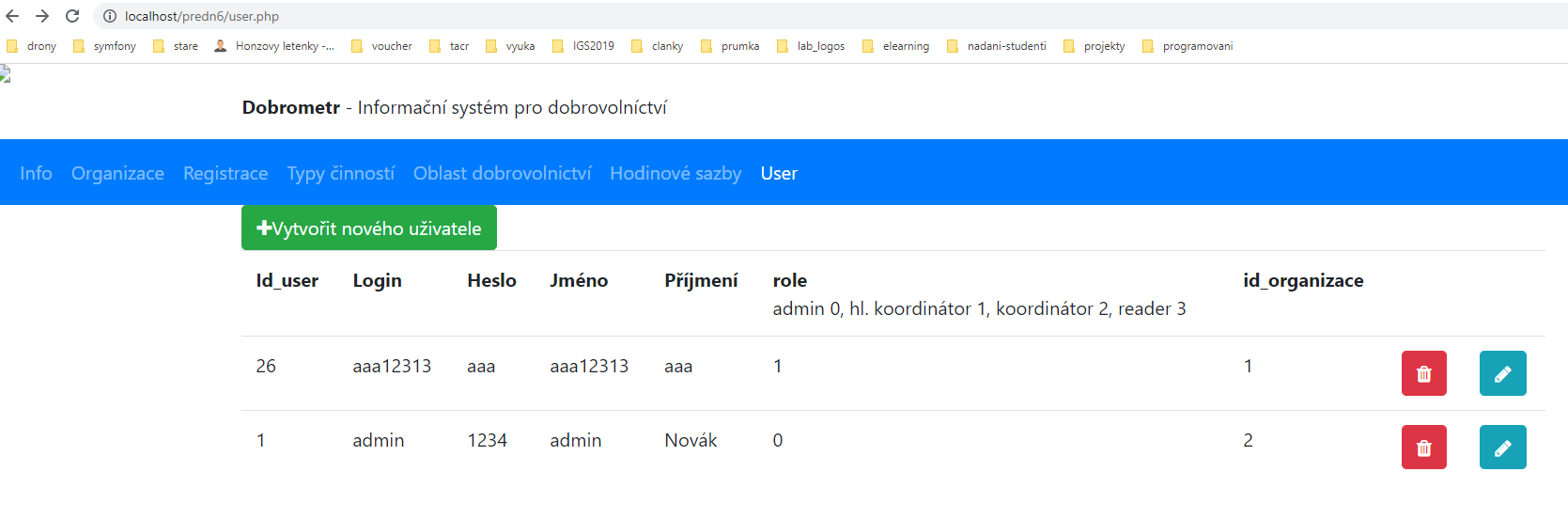
## READALL

<http://localhost/predn6/user/readall.php>



<?php  
*// required headers  
header*("Access-Control-Allow-Origin: \*");  
*header*("Content-Type: application/json; charset=UTF-8");  
   
*// database connection will be here  
  
// include database and object files*include\_once '../config/database.php';  
include\_once '../objects/user.php';  
   
*// instantiate database and product object*$database = new Database();  
$db = $database->getConnection();  
   
*// initialize object*$user = new User($db);  
   
*// read products will be here  
// query products*$stmt = $user->read\_all();  
$num = $stmt->rowCount();  
   
*// check if more than 0 record found  
//echo "adsdfs ".$num;*if($num>0){  
   
 *// products array* $user\_arr=array();  
 $user\_arr["records"]=array();  
   
 while ($row = $stmt->fetch(PDO::*FETCH\_ASSOC*)){  
 *// extract row  
 // this will make $row['name'] to  
 // just $name only  
 extract*($row);  
   
 $user\_item=array(  
 "id" => $id,  
 "login" => $login,  
 "heslo" => $heslo,  
 "jmeno" => $jmeno,  
 "prijmeni" => $prijmeni,  
 "token" => $token,  
 "role" => $role,  
 "id\_organizace" => $id\_organizace  
 );  
   
 *array\_push*($user\_arr["records"], $user\_item);  
 }  
   
 *// set response code - 200 OK  
 http\_response\_code*(200);  
   
 *// show products data in json format* echo *json\_encode*($user\_arr);  
}

### Volání Readall API

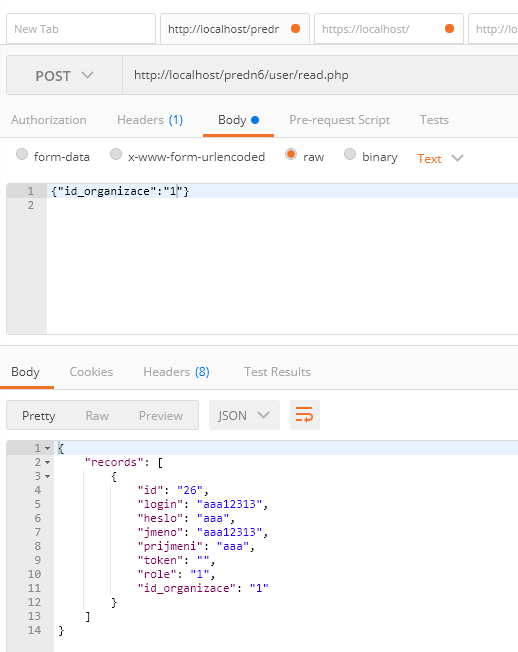


<?php  
*session\_start*();  
  
require('config.php');  
  
if (isset($\_GET["del"])) {  
 $url = $domena."/predn6/user/delete.php";  
 $pole = array('id' => $\_GET["del"]);  
  
 $content = *json\_encode*($pole);  
  
  
 $curl = *curl\_init*($url);  
 *curl\_setopt*($curl, *CURLOPT\_HEADER*, false);  
 *curl\_setopt*($curl, *CURLOPT\_RETURNTRANSFER*, true);  
 *curl\_setopt*($curl, *CURLOPT\_HTTPHEADER*,  
 array("Content-type: application/json"));  
 *curl\_setopt*($curl, *CURLOPT\_POST*, true);  
 *curl\_setopt*($curl, *CURLOPT\_POSTFIELDS*, $content);  
  
 $json\_response = *curl\_exec*($curl);  
  
 $status = *curl\_getinfo*($curl, *CURLINFO\_HTTP\_CODE*);  
  
 *curl\_close*($curl);  
  
*//$response = json\_decode($status, true);* $pole = *explode*(":", $json\_response);  
  
 $vysledekmazani = *mb\_substr*($pole[2], 1, *mb\_strlen*($pole[2]) - 3);  
  
}  
?>  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <title>Uživatel</title>  
 <meta charset="utf-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <link rel="stylesheet" href="bootstrap/css/bootstrap.css">  
 <script src="bootstrap/js/bootstrap.js"></script>  
 <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">  
</head>  
<body>  
  
<img style="float: left" src="ikonky/logo.png">  
<div class="container">  
   
 <p> <br><b>Dobrometr </b>- Informační systém pro dobrovolníctví  
   
   
   
 </p>  
  
</div>  
<div id="header">  
   
<nav class="navbar navbar-expand-sm bg-primary navbar-dark">  
 <ul class="navbar-nav">  
 <li class="nav-item">  
 <a class="nav-link" href="main.php">Info</a>  
 </li>  
   
   
   
   
  
  
 <li class="nav-item ">  
 <a class="nav-link" href="organizacea.php">Organizace</a>  
 </li>  
   
 <li class="nav-item">  
 <a class="nav-link" href="registrace.php">Registrace</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="typaktivity.php">Typy činností</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="kategorie.php">Oblast dobrovolnictví</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="hodinovesazby.php">Hodinové sazby</a>  
 </li>  
 <li class="nav-item active">  
 <a class="nav-link" href="user.php">User</a>  
 </li>   
   
   
  
  
 </ul>  
   
  
</nav>  
 <div class="container">  
   
 <a href="user/new.php"class="btn btn-success"><span class="fa fa-plus"></span>Vytvořit nového uživatele</a>  
 <table class="table">  
 <tr><th>Id\_user</th><th>Login</th><th>Heslo</th><th>Jméno</th><th>Příjmení </th><td><b>role</b><br>admin 0, hl. koordinátor 1, koordinátor 2, reader 3</td><th>id\_organizace</th><th></th><th></th></tr>  
 <?php  
 $data\_array = array("");  
   
 $url = $domena."/predn6/user/readall.php";  
   
 $data = callAPI('POST', $url, *json\_encode*($data\_array));  
  
 $data = *json\_decode*($data, true);  
  
  
 *//var\_dump("volam api: ".$data);  
  
  
  
 // Method: POST, PUT, GET etc  
 // Data: array("param" => "value") ==> index.php?param=value* function CallAPI($method, $url, $data = false)  
 {  
 $curl = *curl\_init*();  
  
 switch ($method)  
 {  
 case "POST":  
 *curl\_setopt*($curl, *CURLOPT\_POST*, 1);  
  
 if ($data)  
 *curl\_setopt*($curl, *CURLOPT\_POSTFIELDS*, $data);  
 break;  
 case "PUT":  
 *curl\_setopt*($curl, *CURLOPT\_PUT*, 1);  
 break;  
 default:  
 if ($data)  
 $url = *sprintf*("%s?%s", $url, *http\_build\_query*($data));  
 }  
  
 *// Optional Authentication:  
 curl\_setopt*($curl, *CURLOPT\_HTTPAUTH*, *CURLAUTH\_BASIC*);  
 *curl\_setopt*($curl, *CURLOPT\_USERPWD*, "username:password");  
  
 *curl\_setopt*($curl, *CURLOPT\_URL*, $url);  
 *curl\_setopt*($curl, *CURLOPT\_RETURNTRANSFER*, 1);  
  
 $result = *curl\_exec*($curl);  
  
 *curl\_close*($curl);  
  
 return $result;  
 }  
  
  
  
 if (isset($data))  
 {  
  
 foreach($data as $d) {  
 foreach ($d as $d2) {  
 *//var\_dump($d2);  
 // var\_dump($d2->id\_kategorie);  
 //var\_dump($d2->oznaceni);* echo "<tr><td>" . $d2['id'] . "</td><td>" . $d2['login'] . "</td><td>" . $d2['heslo'] . "</td><td>" . $d2['jmeno']. "</td><td>" . $d2['prijmeni'] . "</td><td>" . $d2['role']. "</td><td>" . $d2['id\_organizace']. "</td>  
 <td><a href=\"?del=" .$d2['id'] . "\" class=\"btn btn-danger\" title=\"Smazat pracovníka\" onclick=\"return *confirm*('Opravdu chcete pracovníka smazat?')\" ><span class=\"fa fa-trash\"></span> </a></td>";  
  
 echo "<td><a href=\"user/edit.php?id=". $d2['id']."&login=".$d2['login']."&heslo=".$d2['heslo']."&jmeno=".$d2['jmeno']."&prijmeni=".$d2['prijmeni']."&role=".$d2['role']."&id\_organizace=".$d2['id\_organizace']."\" class=\"btn btn-info\" title=\"Editovat\"><span class=\"fa fa-pencil\"></span></a></td>  
 </tr>";  
 }  
 }  
 }  
 ?>  
 </table>  
   
  
  
 </div>  
  
  
  
</body>  
</html>

## READ (uživatel z konkrétní organizace)

<http://localhost/predn6/user/read.php>

{"id\_organizace":"1"}

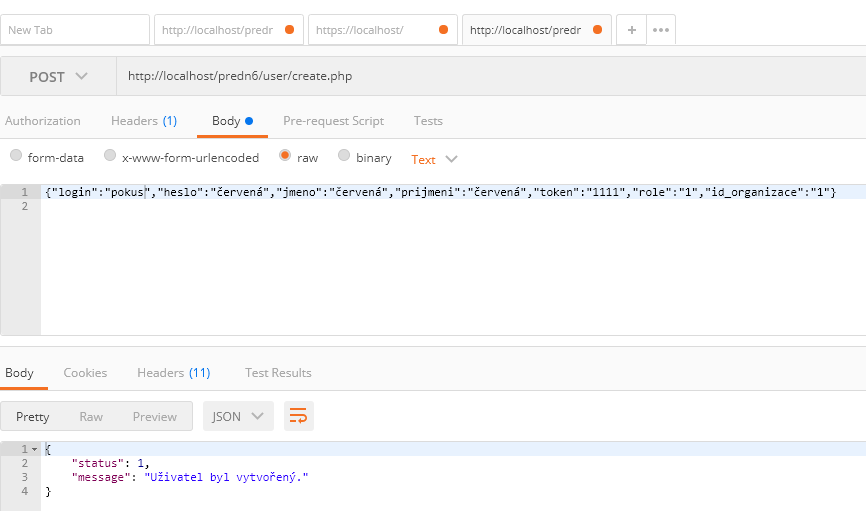


<?php  
  
*// required headers  
header*("Access-Control-Allow-Origin: \*");  
*header*("Content-Type: application/json; charset=UTF-8");  
  
*// database connection will be here  
  
// include database and object files*include\_once '../config/database.php';  
include\_once '../objects/user.php';  
  
*// instantiate database and product object*$database = new Database();  
$db = $database->getConnection();  
  
*// initialize object*$user = new User($db);  
  
*// read products will be here  
// query products*$data = *json\_decode*(*file\_get\_contents*("php://input"));  
  
  
$stmt = $user->read($data->id\_organizace);  
$num = $stmt->rowCount();  
  
*// check if more than 0 record found  
//echo "adsdfs ".$num;*if ($num > 0) {  
  
 *// products array* $user\_arr = array();  
 $user\_arr["records"] = array();  
  
 while ($row = $stmt->fetch(PDO::*FETCH\_ASSOC*)) {  
 *// extract row  
 // this will make $row['name'] to  
 // just $name only  
 extract*($row);  
  
 $user\_item = array(  
 "id" => $id,  
 "login" => $login,  
 "heslo" => $heslo,  
 "jmeno" => $jmeno,  
 "prijmeni" => $prijmeni,  
 "token" => $token,  
 "role" => $role,  
 "id\_organizace" => $id\_organizace  
 );  
  
 *array\_push*($user\_arr["records"], $user\_item);  
 }  
  
 *// set response code - 200 OK  
 http\_response\_code*(200);  
  
 *// show products data in json format* echo *json\_encode*($user\_arr);  
}

## Create

<http://localhost/predn6/user/create.php>

{"login":"Tatra132213","heslo":"červená","jmeno":"červená","prijmeni":"červená","token":"1111","role":"1","id\_organizace":"1"}

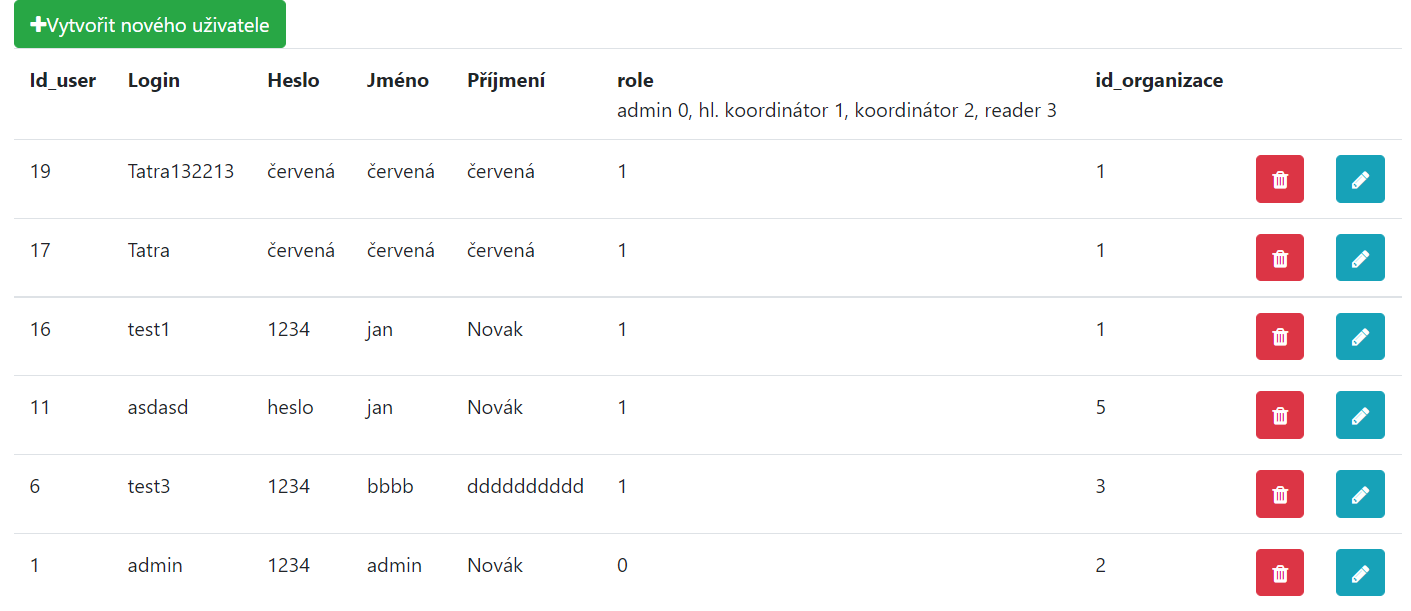


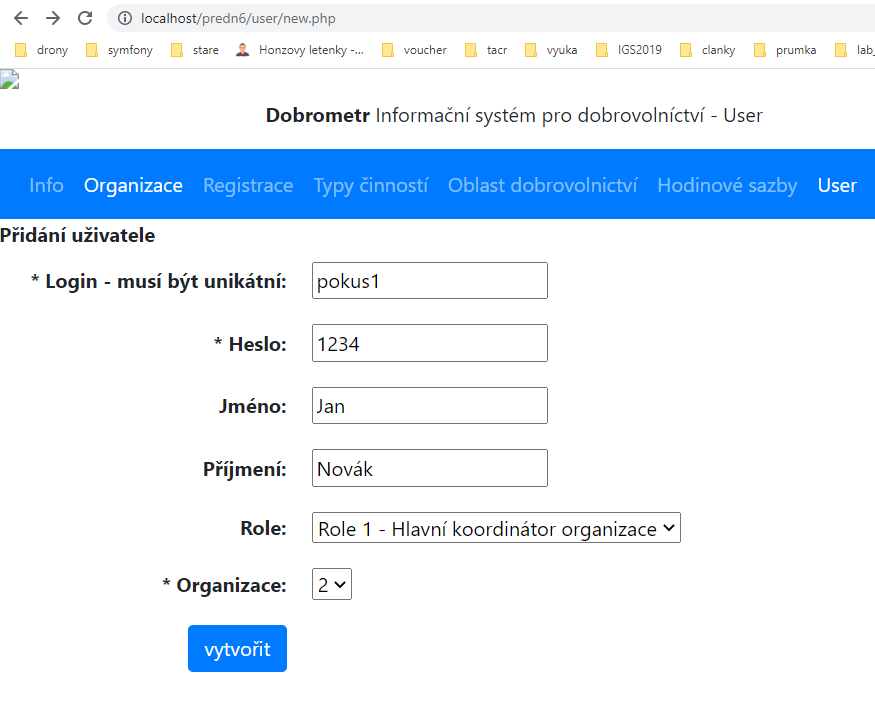
<?php  
*// required headers  
header*("Access-Control-Allow-Origin: \*");  
*header*("Content-Type: application/json; charset=UTF-8");  
*header*("Access-Control-Allow-Methods: POST");  
*header*("Access-Control-Max-Age: 3600");  
*header*("Access-Control-Allow-Headers: Content-Type, Access-Control-Allow-Headers, Authorization, X-Requested-With");  
   
*// include database and object files*include\_once '../config/database.php';  
include\_once '../objects/user.php';  
   
*// get database connection*$database = new Database();  
$db = $database->getConnection();  
   
*// prepare product object*$user = new User($db);  
   
*// get id of product to be edited*$data = *json\_decode*(*file\_get\_contents*("php://input"));  
*//var\_dump($data);  
  
 //echo json\_encode(array("message" => $data->jmeno));   
// make sure data is not empty*if(  
 !empty($data->login)||

){  
   
 *// set product property values* $user->login = $data->login;  
 $user->heslo = $data->heslo;  
 $user->jmeno = $data->jmeno;  
 $user->prijmeni = $data->prijmeni;  
 $user->token = $data->token;  
 $user->role = $data->role;  
 $user->id\_organizace = $data->id\_organizace;  
 *// create the product* if($user->create()){  
   
 *// set response code - 201 created  
 http\_response\_code*(201);  
   
 *// tell the user* echo *json\_encode*(array("status"=>1,"message" => "Uživatel byl vytvořený."));  
 }  
   
 *// if unable to create the user, tell the user* else{  
   
 *// set response code - 503 service unavailable  
 http\_response\_code*(200);  
   
 *// tell the user* echo *json\_encode*(array("status"=>0,"message" => "Uživatel nebyl vytvořený, název není unikátní"));  
 }  
}  
   
*// tell the user data is incomplete*else{  
   
 *// set response code - 400 bad request  
 http\_response\_code*(400);  
   
 *// tell the user* echo *json\_encode*(array("status"=>0,"message" => "Nepovedlo se vytvořit užvatele. Data nejsou kompletní."));  
 }  
?>

### Volání API create

<http://localhost/predn6/user.php>





<http://localhost/predn6/user/new.php>

<?php

require('../config.php');

if (isset($\_POST["poslano"])) {

$url = $domena."/predn6/user/create.php";

$pole = array('login' => $\_POST["login"],'heslo' => $\_POST["heslo"],'jmeno' => $\_POST["jmeno"],'prijmeni' => $\_POST["prijmeni"],'token' => '1','role' => $\_POST["role"],'id\_organizace' => $\_POST["id\_organizace"] );

$content = json\_encode($pole);

$curl = curl\_init($url);

curl\_setopt($curl, CURLOPT\_HEADER, false);

curl\_setopt($curl, CURLOPT\_RETURNTRANSFER, true);

curl\_setopt($curl, CURLOPT\_HTTPHEADER,

array("Content-type: application/json"));

curl\_setopt($curl, CURLOPT\_POST, true);

curl\_setopt($curl, CURLOPT\_POSTFIELDS, $content);

$json\_response = curl\_exec($curl);

$status = curl\_getinfo($curl, CURLINFO\_HTTP\_CODE);

if ($status != 201) {

die("Error: call to URL $url failed with status $status, response $json\_response, curl\_error " . curl\_error($curl) . ", curl\_errno " . curl\_errno($curl));

}

curl\_close($curl);

$response = json\_decode($json\_response, true);

echo $response["message"];

header("Location: ../user.php");

}

?>

<!DOCTYPE html>

<html>

<head lang="cs">

<meta charset="UTF-8">

<title>User - přidat</title>

<link rel="stylesheet" href="../bootstrap/css/bootstrap.css">

<script src="../bootstrap/js/bootstrap.js"></script>

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">

</head>

<body>

<img style="float: left" src="../ikonky/logo.png">

<div class="container">

<br>

<p><b>Dobrometr </b>Informační systém pro dobrovolníctví - User</p>

</div>

<div id="header">

<nav class="navbar navbar-expand-sm bg-primary navbar-dark">

<ul class="navbar-nav">

<li class="nav-item">

<a class="nav-link" href="../main.php">Info</a>

</li>

<li class="nav-item active">

<a class="nav-link" href="../organizacea.php">Organizace</a>

</li>

<li class="nav-item">

<a class="nav-link" href="../registrace.php">Registrace</a>

</li>

<li class="nav-item">

<a class="nav-link" href="../typaktivity.php">Typy činností</a>

</li>

<li class="nav-item">

<a class="nav-link" href="../kategorie.php">Oblast dobrovolnictví</a>

</li>

<li class="nav-item">

<a class="nav-link" href="../hodinovesazby.php">Hodinové sazby</a>

</li>

<li class="nav-item active">

<a class="nav-link" href="../user.php">User</a>

</li>

</ul>

</nav>

<form method="POST" id="zakladni\_info">

<div id="form\_sekce1">

<b>Přidání uživatele</b>

<table cellpadding="10px" width="920px">

<tr>

<td width="240" style="text-align: right"><b><povinne>\*</povinne> Login - musí být unikátní:</b></td>

<td><input type="text" name="login" required></td>

</tr>

<tr>

<td width="240" style="text-align: right"><b><povinne>\*</povinne> Heslo:</b></td>

<td><input type="text" name="heslo" required></td>

</tr><tr>

<td width="240" style="text-align: right"><b> Jméno:</b></td>

<td><input type="text" name="jmeno" required></td>

</tr>

<tr>

<td width="240" style="text-align: right"><b> Příjmení:</b></td>

<td><input type="text" name="prijmeni" required></td>

</tr>

<tr>

<td width="240" style="text-align: right"><b> Role:</b></td>

<td>

<select name="role">

<option value="1">Role 1 - Hlavní koordinátor organizace</option>

<option value="2">Role 2 - Koordinátor organizace</option>

<option value="3">Role 3 - Reader organizace</option>

</select>

</td>

</tr>

<tr>

<td width="240" style="text-align: right"><b><povinne>\*</povinne> Organizace:</b></td>

<td>

<select size="1" name="id\_organizace">

<option value="1"> 1</option>

<option value="2"> 2</option>

</select>

</td>

</tr>

</td></tr>

<tr>

<input type="hidden" name="poslano" value="true">

</tr>

<tr>

<td width="240" style="text-align: right"><input class="btn btn-primary" type="submit" value="vytvořit"></td>

</tr>

</table>

</form>

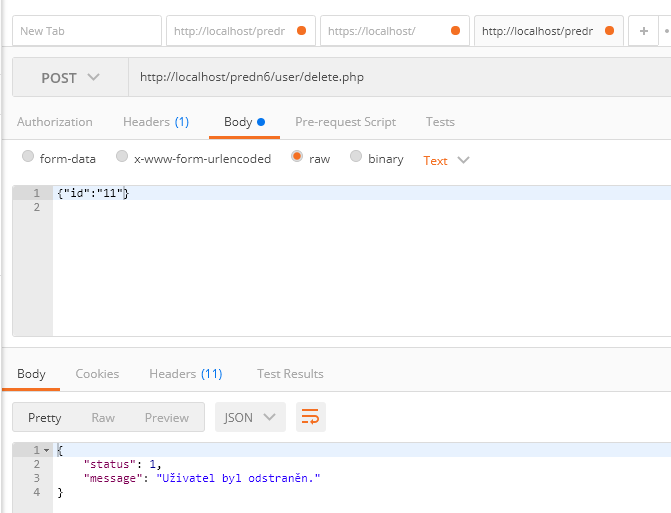
</body>

</html>

## Delete

<http://localhost/predn6/user/delete.php>

{"id":"11"}



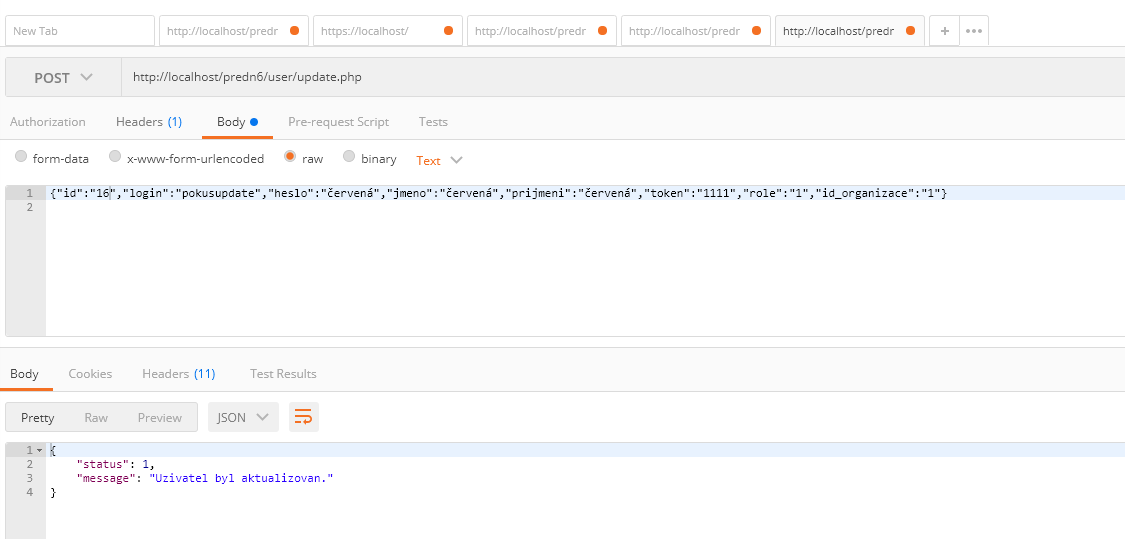
delete.php

<?php  
*// required headers  
header*("Access-Control-Allow-Origin: \*");  
*header*("Content-Type: application/json; charset=UTF-8");  
*header*("Access-Control-Allow-Methods: POST");  
*header*("Access-Control-Max-Age: 3600");  
*header*("Access-Control-Allow-Headers: Content-Type, Access-Control-Allow-Headers, Authorization, X-Requested-With");  
   
*// include database and object file*include\_once '../config/database.php';  
include\_once '../objects/user.php';  
   
*// get database connection*$database = new Database();  
$db = $database->getConnection();  
   
*// prepare product object*$user = new User($db);  
   
*// get product id*$data = *json\_decode*(*file\_get\_contents*("php://input"));  
   
*// set product id to be deleted*$user->id = $data->id;  
   
*// delete the product*if($user->delete()){  
   
 *// set response code - 200 ok  
 http\_response\_code*(200);  
   
 *// tell the user* echo *json\_encode*(array("status"=>1,"message" => "Uživatel byl odstraněn."));  
}  
   
*// if unable to delete the product*else{  
   
 *// set response code - 503 service unavailable  
 http\_response\_code*(503);  
   
 *// tell the user* echo *json\_encode*(array("status"=>0,"message" => "Uživatele nelze smazat."));  
}  
?>

### Volání api delete

<?php  
*session\_start*();  
  
require('config.php');  
  
if (isset($\_GET["del"])) {  
 $url = $domena."/predn6/user/delete.php";  
 $pole = array('id' => $\_GET["del"]);  
  
 $content = *json\_encode*($pole);  
  
  
 $curl = *curl\_init*($url);  
 *curl\_setopt*($curl, *CURLOPT\_HEADER*, false);  
 *curl\_setopt*($curl, *CURLOPT\_RETURNTRANSFER*, true);  
 *curl\_setopt*($curl, *CURLOPT\_HTTPHEADER*,  
 array("Content-type: application/json"));  
 *curl\_setopt*($curl, *CURLOPT\_POST*, true);  
 *curl\_setopt*($curl, *CURLOPT\_POSTFIELDS*, $content);  
  
 $json\_response = *curl\_exec*($curl);  
  
 $status = *curl\_getinfo*($curl, *CURLINFO\_HTTP\_CODE*);  
  
 *curl\_close*($curl);  
  
*//$response = json\_decode($status, true);* $pole = *explode*(":", $json\_response);  
  
 $vysledekmazani = *mb\_substr*($pole[2], 1, *mb\_strlen*($pole[2]) - 3);  
  
}  
?>  
<!DOCTYPE html>  
<html lang="en">  
<head>  
 <title>Uživatel</title>  
 <meta charset="utf-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <link rel="stylesheet" href="bootstrap/css/bootstrap.css">  
 <script src="bootstrap/js/bootstrap.js"></script>  
 <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">  
</head>  
<body>  
  
<img style="float: left" src="ikonky/logo.png">  
<div class="container">  
   
 <p> <br><b>Dobrometr </b>- Informační systém pro dobrovolníctví  
   
   
   
 </p>  
  
</div>  
<div id="header">  
   
<nav class="navbar navbar-expand-sm bg-primary navbar-dark">  
 <ul class="navbar-nav">  
 <li class="nav-item">  
 <a class="nav-link" href="main.php">Info</a>  
 </li>  
   
   
   
   
  
  
 <li class="nav-item ">  
 <a class="nav-link" href="organizacea.php">Organizace</a>  
 </li>  
   
 <li class="nav-item">  
 <a class="nav-link" href="registrace.php">Registrace</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="typaktivity.php">Typy činností</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="kategorie.php">Oblast dobrovolnictví</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="hodinovesazby.php">Hodinové sazby</a>  
 </li>  
 <li class="nav-item active">  
 <a class="nav-link" href="user.php">User</a>  
 </li>   
   
   
  
  
 </ul>  
   
  
</nav>  
 <div class="container">  
   
 <a href="user/new.php"class="btn btn-success"><span class="fa fa-plus"></span>Vytvořit nového uživatele</a>  
 <table class="table">  
 <tr><th>Id\_user</th><th>Login</th><th>Heslo</th><th>Jméno</th><th>Příjmení </th><td><b>role</b><br>admin 0, hl. koordinátor 1, koordinátor 2, reader 3</td><th>id\_organizace</th><th></th><th></th></tr>  
 <?php  
 $data\_array = array("");  
   
 $url = $domena."/predn6/user/readall.php";  
   
 $data = callAPI('POST', $url, *json\_encode*($data\_array));  
  
 $data = *json\_decode*($data, true);  
  
  
 *//var\_dump("volam api: ".$data);  
  
  
  
 // Method: POST, PUT, GET etc  
 // Data: array("param" => "value") ==> index.php?param=value* function CallAPI($method, $url, $data = false)  
 {  
 $curl = *curl\_init*();  
  
 switch ($method)  
 {  
 case "POST":  
 *curl\_setopt*($curl, *CURLOPT\_POST*, 1);  
  
 if ($data)  
 *curl\_setopt*($curl, *CURLOPT\_POSTFIELDS*, $data);  
 break;  
 case "PUT":  
 *curl\_setopt*($curl, *CURLOPT\_PUT*, 1);  
 break;  
 default:  
 if ($data)  
 $url = *sprintf*("%s?%s", $url, *http\_build\_query*($data));  
 }  
  
 *// Optional Authentication:  
 curl\_setopt*($curl, *CURLOPT\_HTTPAUTH*, *CURLAUTH\_BASIC*);  
 *curl\_setopt*($curl, *CURLOPT\_USERPWD*, "username:password");  
  
 *curl\_setopt*($curl, *CURLOPT\_URL*, $url);  
 *curl\_setopt*($curl, *CURLOPT\_RETURNTRANSFER*, 1);  
  
 $result = *curl\_exec*($curl);  
  
 *curl\_close*($curl);  
  
 return $result;  
 }  
  
  
  
 if (isset($data))  
 {  
  
 foreach($data as $d) {  
 foreach ($d as $d2) {  
 *//var\_dump($d2);  
 // var\_dump($d2->id\_kategorie);  
 //var\_dump($d2->oznaceni);* echo "<tr><td>" . $d2['id'] . "</td><td>" . $d2['login'] . "</td><td>" . $d2['heslo'] . "</td><td>" . $d2['jmeno']. "</td><td>" . $d2['prijmeni'] . "</td><td>" . $d2['role']. "</td><td>" . $d2['id\_organizace']. "</td>  
 <td><a href=\"?del=" .$d2['id'] . "\" class=\"btn btn-danger\" title=\"Smazat pracovníka\" onclick=\"return *confirm*('Opravdu chcete pracovníka smazat?')\" ><span class=\"fa fa-trash\"></span> </a></td>";  
  
 echo "<td><a href=\"user/edit.php?id=". $d2['id']."&login=".$d2['login']."&heslo=".$d2['heslo']."&jmeno=".$d2['jmeno']."&prijmeni=".$d2['prijmeni']."&role=".$d2['role']."&id\_organizace=".$d2['id\_organizace']."\" class=\"btn btn-info\" title=\"Editovat\"><span class=\"fa fa-pencil\"></span></a></td>  
 </tr>";  
 }  
 }  
 }  
 ?>  
 </table>  
   
  
  
 </div>  
  
  
  
</body>  
</html>

## Update

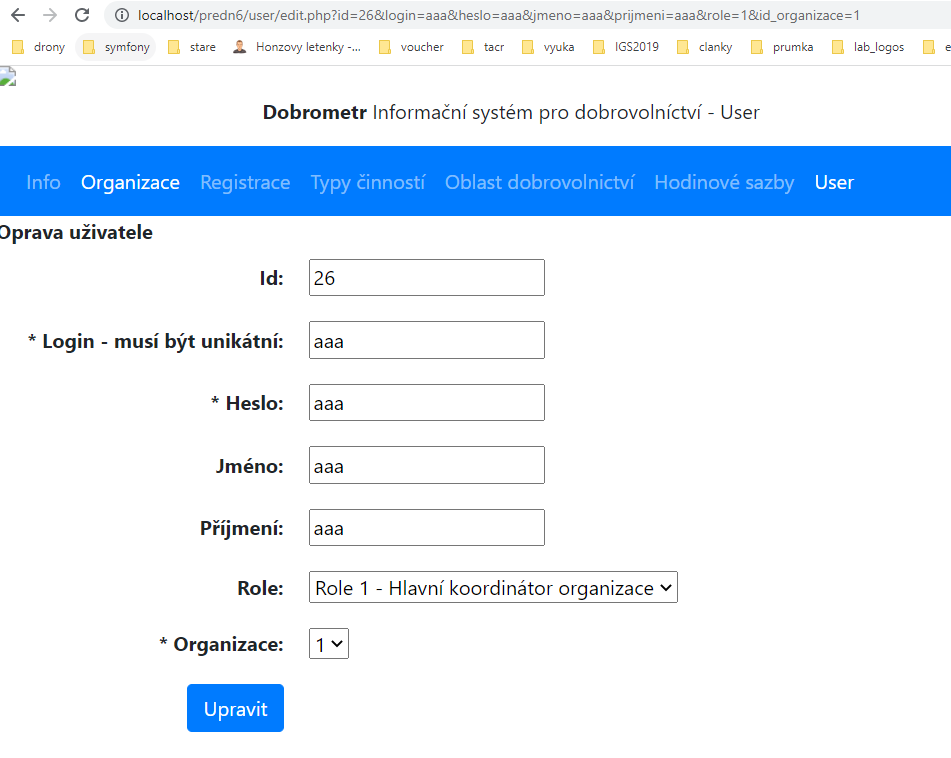


<http://localhost/predn6/user/update.php>

{"id":"16","login":"pokusupdate","heslo":"červená","jmeno":"červená","prijmeni":"červená","token":"1111","role":"1","id\_organizace":"1"}

<?php  
*// required headers  
header*("Access-Control-Allow-Origin: \*");  
*header*("Content-Type: application/json; charset=UTF-8");  
*header*("Access-Control-Allow-Methods: POST");  
*header*("Access-Control-Max-Age: 3600");  
*header*("Access-Control-Allow-Headers: Content-Type, Access-Control-Allow-Headers, Authorization, X-Requested-With");  
   
*// include database and object files*include\_once '../config/database.php';  
include\_once '../objects/user.php';  
   
*// get database connection*$database = new Database();  
$db = $database->getConnection();  
   
*// prepare product object*$user = new User($db);  
   
*// get id of product to be edited*$data = *json\_decode*(*file\_get\_contents*("php://input"));  
   
*// set ID property of product to be edited*$user->id = $data->id;  
$user->login = $data->login;  
$user->heslo = $data->heslo;  
$user->jmeno = $data->jmeno;  
$user->prijmeni = $data->prijmeni;  
$user->id\_organizace = $data->id\_organizace;  
$user->token = "";  
$user->role = $data->role;  
  
   
*// set product property values*if($user->update()){  
  
 *// set response code - 200 ok  
 http\_response\_code*(200);  
  
 *// tell the user* echo *json\_encode*(array("status"=>1,"message" => "Uzivatel byl aktualizovan."));  
}  
  
*// if unable to update the product, tell the user*else{  
  
 *// set response code - 503 service unavailable  
 http\_response\_code*(503);  
  
 *// tell the user* echo *json\_encode*(array("status"=>0,"message" => "Uzivatel nebyl aktualizovan."));  
}  
?>

### Volání API Update



<?php  
  
*session\_start*();  
  
require('../config.php');  
  
if (isset($\_POST["poslano"])) {  
  
 $pole = array('id' => $\_GET["id"],'login' => $\_POST["login"],'heslo' => $\_POST["heslo"],'jmeno' => $\_POST["jmeno"],'prijmeni' => $\_POST["prijmeni"],'role' => $\_POST["role"],'id\_organizace' => $\_POST["id\_organizace"]);  
  
 $url = $domena."/predn6/user/update.php";  
  
  
 $content = *json\_encode*($pole);  
*//var\_dump($content);* $curl = *curl\_init*($url);  
 *curl\_setopt*($curl, *CURLOPT\_HEADER*, false);  
 *curl\_setopt*($curl, *CURLOPT\_RETURNTRANSFER*, true);  
 *curl\_setopt*($curl, *CURLOPT\_HTTPHEADER*,  
 array("Content-type: application/json"));  
 *curl\_setopt*($curl, *CURLOPT\_POST*, true);  
 *curl\_setopt*($curl, *CURLOPT\_POSTFIELDS*, $content);  
  
 $vysledek = *curl\_exec*($curl);  
  
 $status = *curl\_getinfo*($curl, *CURLINFO\_HTTP\_CODE*);  
  
*//if ( $status != 200 ) {  
 // die("Error: call to URL $url failed with status $status, response $json\_response, curl\_error " . curl\_error($curl) . ", curl\_errno " . curl\_errno($curl));  
//}  
  
  
 curl\_close*($curl);  
  
*//$response = json\_decode($json\_response, true);  
//echo $response;  
  
  
//echo(addslashes(json\_encode($vysledek)));* $obj=*json\_encode*($vysledek);  
 $pole=*explode*(":",$vysledek);  
  
 echo *mb\_substr*($pole[2], 1, *mb\_strlen*($pole[1])-3);  
*//var\_dump($vysledek);* echo $response["message"];  
 *header*("Location: ../user.php");  
}  
  
?>  
  
  
  
<!DOCTYPE html>  
<html>  
<head lang="cs">  
 <meta charset="UTF-8">  
 <title>User - přidat</title>  
  
  
 <link rel="stylesheet" href="../bootstrap/css/bootstrap.css">  
 <script src="../bootstrap/js/bootstrap.js"></script>  
  
 <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">  
  
 </head>  
<body>  
<img style="float: left" src="../ikonky/logo.png">  
<div class="container">  
 <br>  
  
 <p><b>Dobrometr </b>Informační systém pro dobrovolníctví - User</p>  
  
</div>  
<div id="header">  
  
  
 <nav class="navbar navbar-expand-sm bg-primary navbar-dark">  
 <ul class="navbar-nav">  
 <li class="nav-item">  
 <a class="nav-link" href="../main.php">Info</a>  
 </li>  
   
   
  
 <li class="nav-item active">  
 <a class="nav-link" href="../organizacea.php">Organizace</a>  
 </li>  
   
 <li class="nav-item">  
 <a class="nav-link" href="../registrace.php">Registrace</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="../typaktivity.php">Typy činností</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="../kategorie.php">Oblast dobrovolnictví</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="../hodinovesazby.php">Hodinové sazby</a>  
 </li>  
 <li class="nav-item active">  
 <a class="nav-link" href="../user.php">User</a>  
 </li>  
   
   
  
 </ul>  
   
  
</nav>  
<form method="POST" id="zakladni\_info">  
 <div id="form\_sekce1">  
 <b>Oprava uživatele</b>  
 <table cellpadding="10px" width="920px">  
 <tr>  
 <td width="240" style="text-align: right"><b> Id:</b></td>  
 <td><input type="text" name="id\_user" value="<?php if (isset($\_GET['id'])) echo $\_GET['id'] ?>" readonly></td>  
 </tr>  
 <tr>  
 <td width="240" style="text-align: right"><b><povinne>\*</povinne> Login - musí být unikátní:</b></td>  
 <td><input type="text" name="login" value="<?php if (isset($\_GET['login'])) echo $\_GET['login'] ?>" required></td>  
 </tr>  
 <tr>  
 <td width="240" style="text-align: right"><b><povinne>\*</povinne> Heslo:</b></td>  
 <td><input type="text" name="heslo" value="<?php if (isset($\_GET['heslo'])) echo $\_GET['heslo'] ?>" required></td>  
 </tr><tr>  
 <td width="240" style="text-align: right"><b> Jméno:</b></td>  
 <td><input type="text" name="jmeno" value="<?php if (isset($\_GET['jmeno'])) echo $\_GET['jmeno'] ?>" required></td>  
 </tr>  
 <tr>  
 <td width="240" style="text-align: right"><b> Příjmení:</b></td>  
 <td><input type="text" name="prijmeni" value="<?php if (isset($\_GET['prijmeni'])) echo $\_GET['prijmeni'] ?>" required></td>  
 </tr>  
   
 <tr>  
 <td width="240" style="text-align: right"><b> Role:</b></td>  
 <td>  
 <select name="role">  
 <option <?php if ($\_GET['role']=="0" ) echo 'selected' ; ?> value="0">Role 0 - Admin</option>  
 <option <?php if ($\_GET['role']=="1" ) echo 'selected' ; ?> value="1">Role 1 - Hlavní koordinátor organizace</option>  
 <option <?php if ($\_GET['role']=="2" ) echo 'selected' ; ?>value="2">Role 2 - Koordinátor organizace</option>  
 <option <?php if ($\_GET['role']=="3" ) echo 'selected' ; ?>value="2">Role 3 - Reader organizace</option>  
 </select>  
 </td>  
 </tr>  
   
  
   
 <tr>  
 <td width="240" style="text-align: right"><b><povinne>\*</povinne> Organizace:</b></td>  
 <td>  
  
 <select size="1" name="id\_organizace">  
  
 <option value="1"> 1</option>  
 <option value="2"> 2</option>  
   
  
 </select>  
  
   
   
   
 </td>  
 </tr>  
   
   
 </td>  
 </tr>  
 <tr>  
 <input type="hidden" name="poslano" value="true">  
 </tr>  
 <tr>  
 <td width="240" style="text-align: right"><input class="btn btn-primary" type="submit" value="Upravit")></td>  
 </tr>  
 </table>  
</form>  
</body>  
</html>

## Join

<?php

class Product{

// database connection and table name

private $conn;

private $table\_name = "products";

// object properties

public $id;

public $name;

public $description;

public $price;

public $category\_id;

public $category\_name;

public $created;

// constructor with $db as database connection

public function \_\_construct($db){

$this->conn = $db;

}

// read products

function read(){

// select all query

$query = "SELECT

c.name as category\_name, p.id, p.name, p.description, p.price, p.category\_id, p.created

FROM

" . $this->table\_name . " p

LEFT JOIN

categories c

ON p.category\_id = c.id

ORDER BY

p.created DESC";

// prepare query statement

$stmt = $this->conn->prepare($query);

// execute query

$stmt->execute();

return $stmt;

}

// create product

function create(){

// query to insert record

$query = "INSERT INTO

" . $this->table\_name . "

SET

name=:name, price=:price, description=:description, category\_id=:category\_id, created=:created";

// prepare query

$stmt = $this->conn->prepare($query);

// sanitize

$this->name=htmlspecialchars(strip\_tags($this->name));

$this->price=htmlspecialchars(strip\_tags($this->price));

$this->description=htmlspecialchars(strip\_tags($this->description));

$this->category\_id=htmlspecialchars(strip\_tags($this->category\_id));

$this->created=htmlspecialchars(strip\_tags($this->created));

// bind values

$stmt->bindParam(":name", $this->name);

$stmt->bindParam(":price", $this->price);

$stmt->bindParam(":description", $this->description);

$stmt->bindParam(":category\_id", $this->category\_id);

$stmt->bindParam(":created", $this->created);

// execute query

if($stmt->execute()){

return true;

}

return false;

}

// used when filling up the update product form

function readOne(){

// query to read single record

$query = "SELECT

c.name as category\_name, p.id, p.name, p.description, p.price, p.category\_id, p.created

FROM

" . $this->table\_name . " p

LEFT JOIN

categories c

ON p.category\_id = c.id

WHERE

p.id = ?

LIMIT

0,1";

// prepare query statement

$stmt = $this->conn->prepare( $query );

// bind id of product to be updated

$stmt->bindParam(1, $this->id);

// execute query

$stmt->execute();

// get retrieved row

$row = $stmt->fetch(PDO::FETCH\_ASSOC);

// set values to object properties

$this->name = $row['name'];

$this->price = $row['price'];

$this->description = $row['description'];

$this->category\_id = $row['category\_id'];

$this->category\_name = $row['category\_name'];

}

// update the product

function update(){

// update query

$query = "UPDATE

" . $this->table\_name . "

SET

name = :name,

price = :price,

description = :description,

category\_id = :category\_id

WHERE

id = :id";

// prepare query statement

$stmt = $this->conn->prepare($query);

// sanitize

$this->name=htmlspecialchars(strip\_tags($this->name));

$this->price=htmlspecialchars(strip\_tags($this->price));

$this->description=htmlspecialchars(strip\_tags($this->description));

$this->category\_id=htmlspecialchars(strip\_tags($this->category\_id));

$this->id=htmlspecialchars(strip\_tags($this->id));

// bind new values

$stmt->bindParam(':name', $this->name);

$stmt->bindParam(':price', $this->price);

$stmt->bindParam(':description', $this->description);

$stmt->bindParam(':category\_id', $this->category\_id);

$stmt->bindParam(':id', $this->id);

// execute the query

if($stmt->execute()){

return true;

}

return false;

}

// delete the product

function delete(){

// delete query

$query = "DELETE FROM " . $this->table\_name . " WHERE id = ?";

// prepare query

$stmt = $this->conn->prepare($query);

// sanitize

$this->id=htmlspecialchars(strip\_tags($this->id));

// bind id of record to delete

$stmt->bindParam(1, $this->id);

// execute query

if($stmt->execute()){

return true;

}

return false;

}

// search products

function search($keywords){

// select all query

$query = "SELECT

c.name as category\_name, p.id, p.name, p.description, p.price, p.category\_id, p.created

FROM

" . $this->table\_name . " p

LEFT JOIN

categories c

ON p.category\_id = c.id

WHERE

p.name LIKE ? OR p.description LIKE ? OR c.name LIKE ?

ORDER BY

p.created DESC";

// prepare query statement

$stmt = $this->conn->prepare($query);

// sanitize

$keywords=htmlspecialchars(strip\_tags($keywords));

$keywords = "%{$keywords}%";

// bind

$stmt->bindParam(1, $keywords);

$stmt->bindParam(2, $keywords);

$stmt->bindParam(3, $keywords);

// execute query

$stmt->execute();

return $stmt;

}

// read products with pagination

public function readPaging($from\_record\_num, $records\_per\_page){

// select query

$query = "SELECT

c.name as category\_name, p.id, p.name, p.description, p.price, p.category\_id, p.created

FROM

" . $this->table\_name . " p

LEFT JOIN

categories c

ON p.category\_id = c.id

ORDER BY p.created DESC

LIMIT ?, ?";

// prepare query statement

$stmt = $this->conn->prepare( $query );

// bind variable values

$stmt->bindParam(1, $from\_record\_num, PDO::PARAM\_INT);

$stmt->bindParam(2, $records\_per\_page, PDO::PARAM\_INT);

// execute query

$stmt->execute();

// return values from database

return $stmt;

}

// used for paging products

public function count(){

$query = "SELECT COUNT(\*) as total\_rows FROM " . $this->table\_name . "";

$stmt = $this->conn->prepare( $query );

$stmt->execute();

$row = $stmt->fetch(PDO::FETCH\_ASSOC);

return $row['total\_rows'];

}

}

## Zobrazení jednoho řádku bez cyklu

### Objekt aktivita

function souhrny($datumod,$datumdo,$id\_organizace,$id\_projekt)

{

//$datumod = date('YY-m-d', strtotime($datumod));

// $od = new DateTime($datumod);

// $od=$od->format('Y-m-d');

$query = "SELECT

SUM(p.pocet\_osob) as pocetosob, SUM(p.pocet\_hodin) as pocethodin, SUM(p.cena) as cena, (pro.oznaceni) AS nazevprojektu

FROM

" . $this->table\_name . " p

LEFT JOIN

projekt pro

ON p.id\_projekt = pro.id

WHERE

p.datum>=? AND p.datum<=? AND p.id\_organizace=? AND p.id\_projekt=?";

// prepare query statement

$stmt = $this->conn->prepare($query);

$stmt->bindParam(1, $datumod, PDO::PARAM\_INT);

$stmt->bindParam(2, $datumdo, PDO::PARAM\_INT);

$stmt->bindParam(3, $id\_organizace, PDO::PARAM\_INT);

$stmt->bindParam(4, $id\_projekt, PDO::PARAM\_INT);

// execute query

$stmt->execute();

return $stmt;

}

### Zobrazení dat

$pocet\_osob=$pocet\_osob+$data['records'][0]['pocetosob'];

# Zabezpečení

1. posílat token buď v head nebo body
2. JWT <https://codeofaninja.com/2018/09/rest-api-authentication-example-php-jwt-tutorial.html> JSON Web Token
3. [OAuth](https://en.wikipedia.org/wiki/OAuth) nejbezpečnější