Міністерство освіти і науки України  
Івано-Франківський національний технічний університет нафти і газу

Кафедра програмного   
забезпечення автоматизованих систем

Лабораторна робота №5

Розроблення RESTful API для сайту

Виконав: ст. гр. ПІ-13-1

Колесник Андрій

Перевірив:   
Броновський І.В.

м. Івано-Франківськ

2016

**Мета:** Навчитися реалізовувати RESTful API для своїх веб-додатків.

**Завдання**: Розробити API для сайту використовуючи методи POST, GET, PUT, PATCH, DELETE Додати можливість додавання товарів (книжок, людей, згідно вашої теми). Додати можливість додавання, видалення, редагування. Розробити сторінку з документацією доступних методів і їх приклади використання.

**Виконання завдання:**

Для реалізації API в розроблюваній системі було написано окремий контролер.

<?php

class API\_Controller extends Controller {

function user\_data () {

if (array\_key\_exists('id', $this->params)) {

$user = User::find($this->params["id"]);

} else if (array\_key\_exists('email', $this->params)) {

$user = User::where("email='".$this->params["email"]."'").take\_one();

}

if ($user) {

unset($user["password"]);

unset($user["updated\_at"]);

header('Content-type: application/json');

echo json\_encode($user, JSON\_UNESCAPED\_UNICODE | JSON\_PRETTY\_PRINT);

} else {

echo json\_encode("");

}

}

function find\_users () {

if (array\_key\_exists('name', $this->params)) {

$name = mb\_strtolower ($this->params["name"]);

$exploded = explode(' ', $name);

if (count($exploded) > 1) {

$query = User::where("LOWER(first\_name) LIKE '%".$exploded[0]."%'");

$query->where("LOWER(last\_name) LIKE '%".$exploded[1]."%'");

} else {

$query = User::where("LOWER(first\_name) LIKE '%".$exploded[0]."%' OR LOWER(last\_name) LIKE '%".$exploded[0]."%'");

}

$result = $query->take();

}

if ($result) {

header('Content-type: application/json');

echo json\_encode($result, JSON\_UNESCAPED\_UNICODE | JSON\_PRETTY\_PRINT);

} else {

echo json\_encode("");

}

}

function institution\_data () {

if (array\_key\_exists('id', $this->params)) {

$institution = Institution::find($this->params["id"]);

}

if (isset($institution)) {

header('Content-type: application/json');

echo json\_encode($institution, JSON\_UNESCAPED\_UNICODE | JSON\_PRETTY\_PRINT);

} else {

echo json\_encode("");

}

}

function find\_institutions () {

if (array\_key\_exists('name', $this->params)) {

$name = mb\_strtolower ($this->params["name"]);

$exploded = explode(' ', $name);

$query = Institution::where("LOWER(name) LIKE '%".$name."%'");

$result = $query->take();

}

if ($result) {

header('Content-type: application/json');

echo json\_encode($result, JSON\_UNESCAPED\_UNICODE | JSON\_PRETTY\_PRINT);

} else {

echo json\_encode("");

}

}

function login () {

$user = User::where("email='".$this->params["email"]."' AND password='".$this->params["password"]."'")->take\_one();

if ($user) {

$\_SESSION["email"] = $this->params["email"];

$\_SESSION["password"] = $this->params["password"];

echo json\_encode(array('status' => 'success'), JSON\_PRETTY\_PRINT);

} else {

echo json\_encode(array('status' => 'fail'), JSON\_PRETTY\_PRINT);

}

}

function logout () {

$\_SESSION["user"] = null;

$\_SESSION["email"] = null;

$\_SESSION["password"] = null;

echo json\_encode("you have been logged out");

}

function signup () {

if (array\_key\_exists('user', $this->params)) {

$result = User::insert($this->params["user"]);

if ($result) {

echo json\_encode(array('status' => 'success'), JSON\_PRETTY\_PRINT);

} else {

echo json\_encode(array('status' => 'fail'), JSON\_PRETTY\_PRINT);

}

} else {

echo json\_encode(array('status' => 'fail', 'error' => 'user parameter was not specified'));

}

}

function new\_institution () {

if (array\_key\_exists('institution', $this->params)) {

$result = User::insert($this->params["institution"]);

if ($result) {

echo json\_encode(array('status' => 'success'), JSON\_PRETTY\_PRINT);

} else {

echo json\_encode(array('status' => 'fail'), JSON\_PRETTY\_PRINT);

}

} else {

echo json\_encode(array('status' => 'fail', 'error' => 'institution parameter was not specified'));

}

}

function update\_user () {

if (array\_key\_exists('id', $this->params)) {

$user = User::find($this->params["id"]);

}

$curr\_user = $\_SESSION["user"];

if (isset($user) && isset($curr\_user) && ($curr\_user["permissions"] == "admin" || $user["id"] == $curr\_user["id"])) {

$result = User::update($this->params["values"], array('id' => $this->params["id"]));

if ($result) {

echo json\_encode(array('status' => 'success'), JSON\_PRETTY\_PRINT);

} else {

echo json\_encode(array('status' => 'fail'), JSON\_PRETTY\_PRINT);

}

}

}

function update\_institution () {

if (array\_key\_exists('id', $this->params)) {

$institution = Institution::find($this->params["id"]);

}

$user = $\_SESSION["user"];

if (isset($institution) && isset($user) && ($user["permissions"] == "admin" || $user["id"] == $institution["user\_id"])) {

$result = Institution::update($this->params["values"], array('id' => $this->params["id"]));

if ($result) {

echo json\_encode(array('status' => 'success'), JSON\_PRETTY\_PRINT);

} else {

echo json\_encode(array('status' => 'fail'), JSON\_PRETTY\_PRINT);

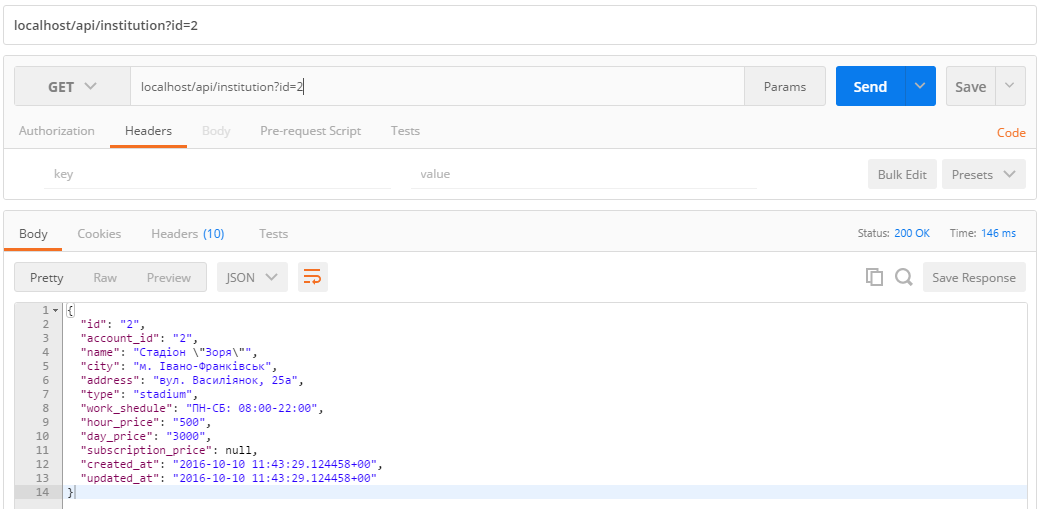
}

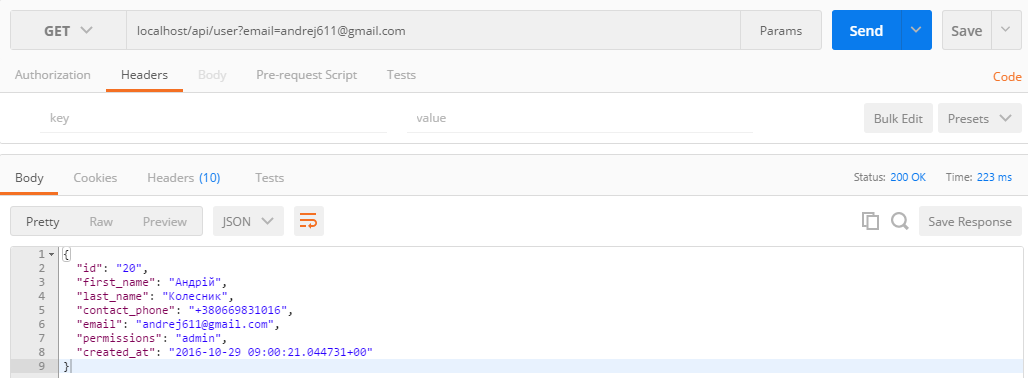
}

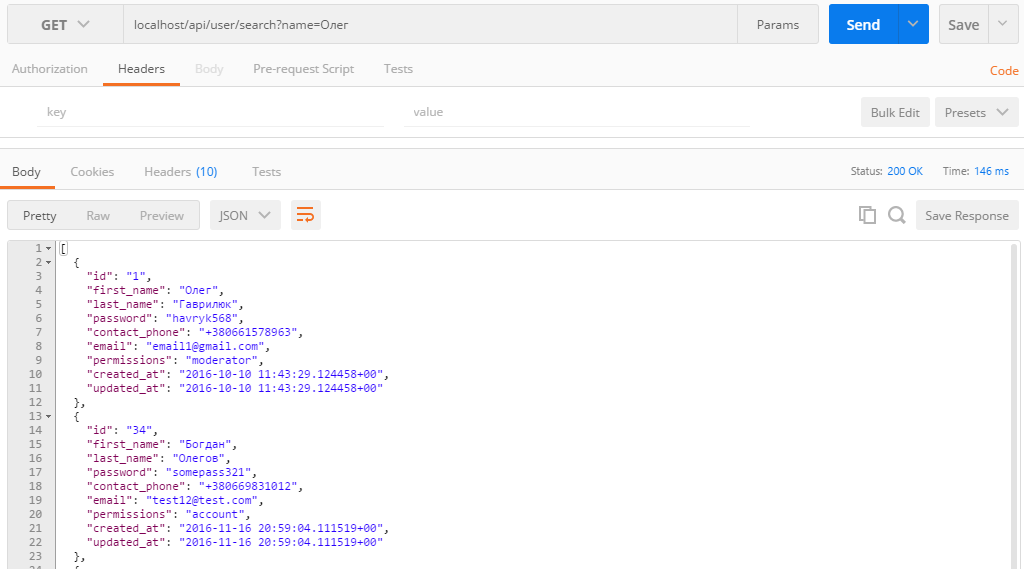
}

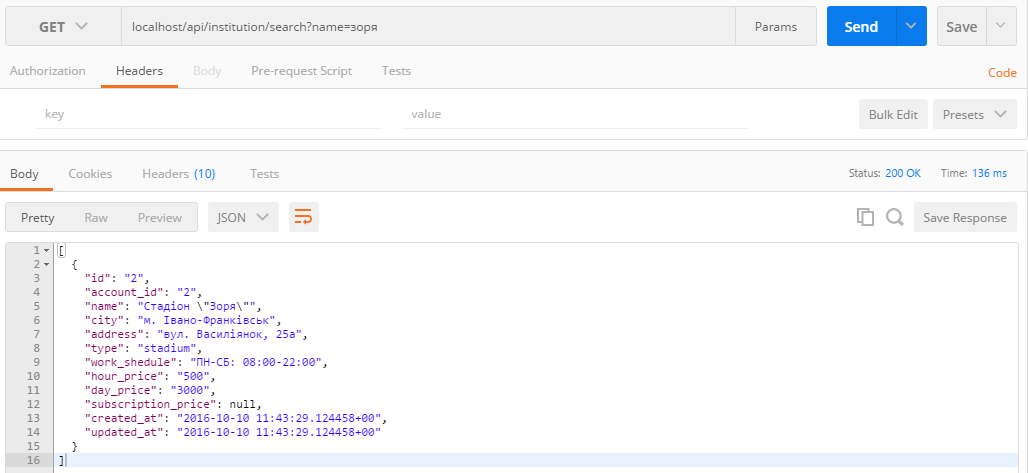
}

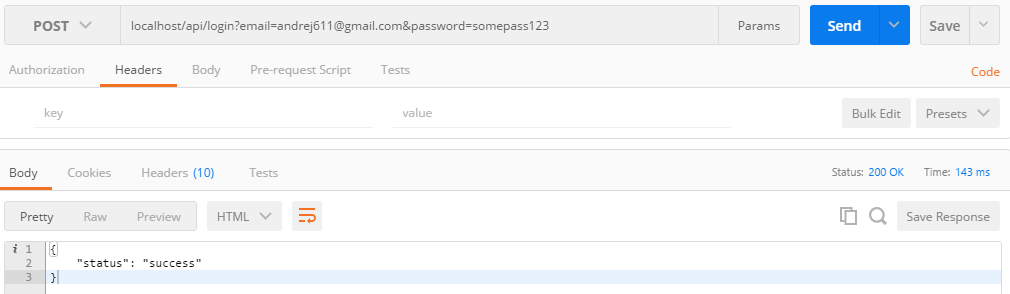
Результати виконання деяких запитів до серверу за допомогою розширення Postman:



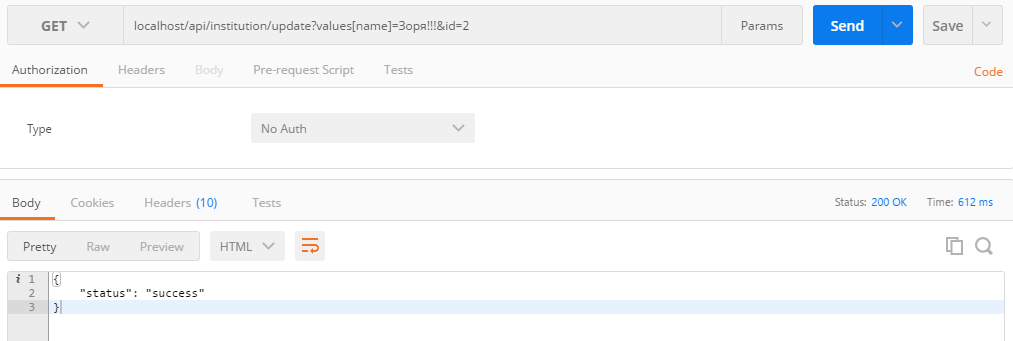


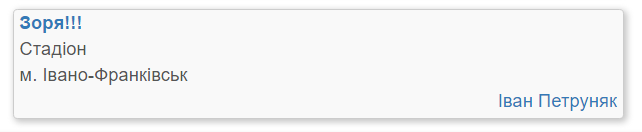






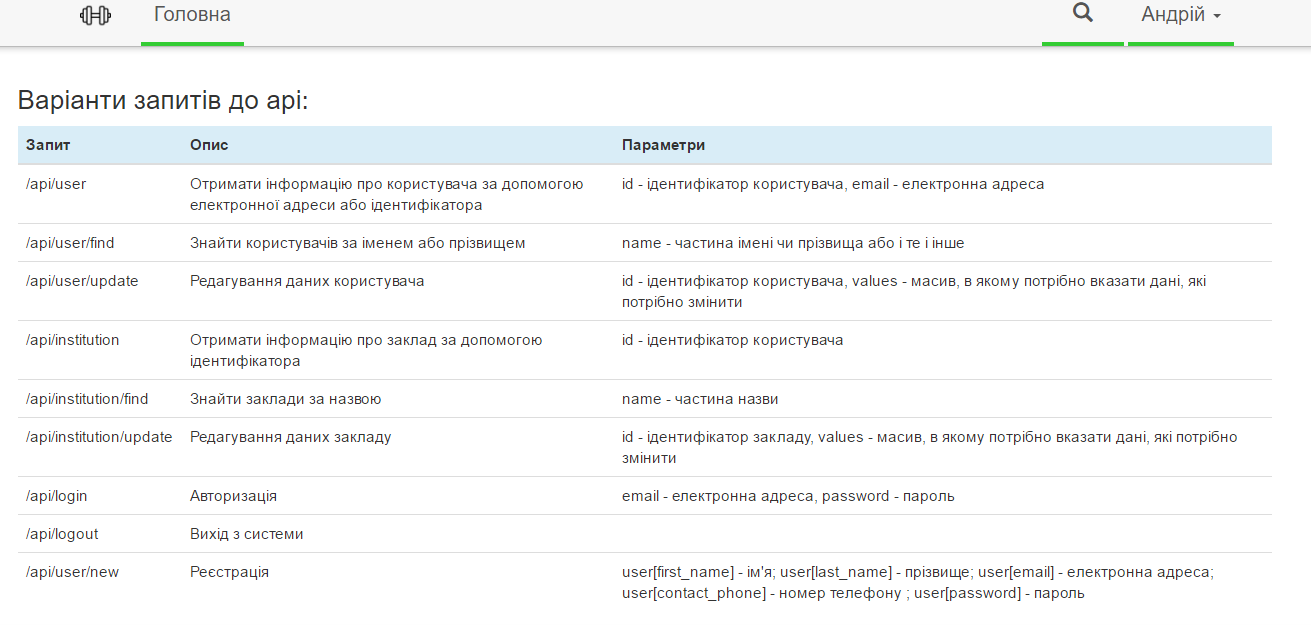
Редагування закладу за допомогою API





Для того, щоб виконати таку дію, користувач повинен бути авторизований в системі, та бути або творцем даного закладу або адміністратором.

Сторінка зі списком команд до api (/api):



**Висновок:** на цій лабораторній роботі було розроблено RESTful API засобами мови PHP.