

# PAŁACE I PARKI WebAPI

Opracowanie: Centrum Technologii Internetowych CTI sp. z o.o.

## I. AUTORYZACJA

Api do autoryzacji wykorzystuje nagłówek **HTTP\_AUTHORIZATION**.

Wysyłając żądanie cURL należy dodać zmienną **CURLOPT\_USERPWD**, która zawiera **login:hasło** użytkownika api.

```
curl_setopt($ch, CURLOPT_USERPWD, 'login:hasło');
```

lub

```
$headers = array(
    'Authorization: Basic '. base64_encode("login:hasło")
);
curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
```

lub zapytaniu konsolowym

```
-H "Authorization: Basic ". base64_encode("login:hasło")
```

## II. METODY

Wszystkie odpowiedzi metod są zwracane w formacie **json**.

### 1. Pałace i zamki – lista

URL żądania: <https://api.palaceiparki.pl/api/places/list/>

Metoda zwraca tablicę (listę) pałaców i zamków bez detali.

Każdy wiersz zawiera następujące informacje:

- **id** - id wpisu
- **sequence** - kolejność według której są wyświetlane na www
- **title** - nazwę
- **autor** - autora wpisu
- **community** - gmina
- **county** - powiat
- **city** - miejscowość
- **image** - tablica ze zdjęciem do wpisu, zawierająca 4 rozmiary, opis

## 2. Pałace i zamki opracowania

URL żądania: <https://api.palaceiparki.pl/api/places/details/>

Metoda zwraca tablicę (listę) szczegółów poszczególnych pałaców i zamków.

Każdy wiersz zawiera następujące informacje:

- **id** - id wpisu
- **sequence** - kolejność według której są wyświetlane na www
- **place\_id** - id wpisu z metody 1, po którym wiążemy dane z metody 1 i 2
- **title** - nazwę
- **contents** - opis
- **notes** - tablica (lista) przypisów, tutaj opis wyświetlania trzeba dodać, jakieś #hasze

## 3. Pałace i zamki ikonografia

URL żądania: <https://api.palaceiparki.pl/api/places/gallery/>

Metoda zwraca tablicę (listę) galerii poszczególnych pałaców i zamków.

Każdy wiersz zawiera następujące informacje:

- **id** - id wpisu
- **sequence** - kolejność według której są wyświetlane na www
- **place\_id** - id wpisu z metody 1, po którym wiążemy dane z metody 1 i 2
- **title** - nazwę
- **contents**- opis
- **images** - tablica (lista) zdjęć, zawierająca 4 rozmiary, opis

## 4. Szlakiem architektury rezydencjonalnej regionu

URL żądania: <https://api.palaceiparki.pl/api/architecture/details/>

Metoda zwraca tablicę (listę) wpisów wraz z dodaną do nich galerią.

Każdy wiersz zawiera następujące informacje:

- **id** - id wpisu
- **sequence** - kolejność według której są wyświetlane na www
- **title** - nazwę
- **contents** - opis
- **images** - tablica (lista) zdjęć, zawierająca 4 rozmiary, opis

## 5. O projekcie

URL żądania: <https://api.palaceiparki.pl/api/project/details/>

Metoda zwraca tablicę (listę) wpisów, podzieloną na 3 działy:

### - Cele i założenia projektu ['goals']

- **title** - tytuł wpisu
- **contents** - treść wpisu

### - Zespół ['team']

- **title** - tytuł wpisu
- **contents** - treść wpisu

### - Plany rozwoju projektu ['plans']

- **title** - tytuł wpisu
- **contents** - treść wpisu
- **rows** - tablica / lista wpisów składających się z:
  - **title** - tytuł
  - **contents** - treść
- **images** - tablica/lista obrazków jw.

## 6. Partnerzy

URL żądania: <https://api.palaceiparki.pl/api/partners/details/>

Metoda zwraca:

- **title** – tytuł wpisu
- **contents** – treść wpisu
- **images** - tablica/lista obrazków jw.
- **partners** - lista wpisów składających się z:
  - **title** – tytuł
  - **add** - dopisek
  - **image** - zdjęcie

## 7. Kontakt

URL żądania: <https://api.palaceiparki.pl/api/contact/details/>

Metoda zwraca:

- **title** – tytuł wpisu
- **link** – link do strony fundacji
- **contents** – treść wpisu

## III. KODY NAGŁÓWKÓW ODPOWIEDZI

- **200** - ok
- **401** - błąd autoryzacji
- **406** - błędna metoda
- **429** - zbyt duża liczba połączeń

## IV. PRZYKŁAD SKRYPTU PHP DO POŁĄCZENIA Z API

```
connect('https://api.palaceiparki.pl/api/places/list/');

function connect($url) {

    $ch = curl_init();
    curl_setopt($ch, CURLOPT_URL, $url);
    /*poniższe linijki tylko w razie problemów z połączeniem*/
    //curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, 0);
```

```
//curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, 0);
/****/
curl_setopt($ch, CURLOPT_HEADER, 0);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
curl_setopt($ch, CURLOPT_TIMEOUT, 15);
curl_setopt($ch, CURLOPT_HTTPAUTH, CURLAUTH_BASIC);
curl_setopt($ch, CURLOPT_USERPWD, 'login:hasło');
$response = curl_exec($ch);
$curl_info = curl_getinfo($ch);
curl_close($ch);

if($curl_info['http_code'] == 200){
    $response = json_decode($response, true);
    var_dump('Header: '.$curl_info['http_code'], $response);
}else{
    var_dump('Header: '.$curl_info['http_code'], $response);
}
}
```