

# Dokumentacja API Bazy Wiedzy

---

## Spis treści

|   |    |
|---|----|
| Wstęp .....   | 5  |
| Format odpowiedzi .....   | 5  |
| Opis funkcji .....  | 6  |
| 1. ApiCommons .....   | 6  |
| 1.1. getVoivodeships - Słownik województwa .....  | 6  |
| 1.2. getPKD - Słownik PKD .....   | 6  |
| 1.3. getPKDDetails - Pobranie szczegółów PKD .....  | 7  |
| 1.4. getPKDList - wyszukiwanie PKD .....  | 9  |
| 1.5. getPKDHints – Pobranie podpowiedzi dla wyszukiwarki procedur po pkd .....                                    | 10 |
| 1.6. getSYM – wyszukanie miejscowości w oparciu o tekst lub kod pocztowy .....                                    | 11 |
| 2. ApiLifeSituations .....  | 13 |
| 2.1. Algorytm .....   | 13 |
| 2.2. getPDF - Pobranie PDF-a .....  | 13 |
| 2.3. startLifeSituation - start sytuacji życiowej .....   | 14 |
| 2.4. getCurrentQuestion - pobranie pytania .....  | 15 |
| 2.5. saveCurrentQuestion – zapisanie bieżącego pytania .....  | 17 |
| 2.6. revertToPreviousQuestion - cofnięcie do poprzedniego pytania .....   | 17 |
| 2.7. revertToStage – powrót do wybranego etapu .....  | 18 |
| 2.8. getEndDescription - pobranie końcowego opisu .....   | 19 |
| 2.9. getSummary - pobranie podsumowania .....   | 20 |
| 2.10. getAnswers - Metoda pobiera odpowiedzi sytuacji życiowej .....  | 20 |
| 2.11. getList – Pobranie listy sytuacji życiowych .....   | 21 |
| 2.12. getDescription Pobranie opisu sytuacji życiowej .....   | 23 |
| 2.13. linkSessionWithUser Powiązanie sesji sytuacji życiowej z użytkownikiem .....                                | 25 |
| 2.14. getVisibleLifeSituationsPropositions Pobranie listy widocznych propozycji sytuacji<br>życiowych .....       | 25 |
| 2.15. voteOnLifeSituationProposition Głosowanie na propozycję sytuacji życiowej .....                             | 26 |
| 3. ApiOpinions .....  | 28 |
| 3.1. saveOpinionForContent - Dodawanie opinii .....   | 28 |
| 4. ApiInstitutions .....  | 30 |
| 4.1. saveInstitutionProposition - Zapisywanie propozycji nowej instytucji .....                                   | 30 |
| 4.2. getListToPerformElectronicProcedure – Pobranie list instytucji wykonującej elektronicznie<br>procedurę ..... | 32 |

|        |   |    |
|--------|---|----|
| 4.3.   | getCategories - Pobranie listy kategorii instytucji .....   | 33 |
| 4.4.   | getHints Pobieranie podpowiedzi nazw instytucji .....   | 33 |
| 4.5.   | getList - Pobieranie listy instytucji .....   | 34 |
| 4.6.   | getDetails - Pobieranie szczegółów instytucji .....   | 37 |
| 4.7.   | getDepartmentDetails - Pobieranie szczegółów wydziału instytucji.....                                   | 38 |
| 4.8.   | getListByTeryt - Pobieranie listy jednostek samorządu terytorialnego .....                              | 40 |
| 5.     | ApiProcedures .....   | 42 |
| 5.1.   | Wprowadzenie .....  | 42 |
| 5.1.1. | Zastosowanie parametru ALIAS zamiast ID w wywołaniach .....   | 42 |
| 5.2.   | getHints - Pobieranie podpowiedzi nazw procedur .....   | 42 |
| 5.3.   | getList - Pobieranie listy procedur .....   | 43 |
| 5.4.   | getElectronicList - Pobieranie listy procedur elektronicznych .....                                     | 46 |
| 5.5.   | getDetails - Pobieranie szczegółów procedury.....   | 48 |
|        | Znaczenia „executing_institution” .....   | 56 |
| 5.6.   | getThematicCategories - Pobieranie katalogu tematycznego.....   | 56 |
| 5.7.   | getProfessionCategories - Pobieranie katalogu zawodów .....   | 57 |
| 5.8.   | getListForRespAuthority - Pobieranie listy procedur dla Instytucji Centralnych /<br>Akceptujących ..... | 57 |
| 6.     | ApiPnaInstitutions .....  | 60 |
| 6.1.   | getHintsByCity – pobieranie podpowiedzi po nazwie miejscowości.....                                     | 60 |
| 6.2.   | getListByLocation – pobranie instytucji na podstawie lokalizacji .....                                  | 61 |
| 7.     | ApiPnaAddress.....  | 63 |
| 7.1.   | getListByPostCode – lista miejscowości po kodzie pocztowym .....  | 63 |
| 7.2.   | getStreetHints - pobranie podpowiedzi ulic.....   | 63 |
| 8.     | ApiPublications .....   | 65 |
| 8.1.   | getDetails – Pobranie publikacji .....  | 65 |
| 8.2.   | getList – Pobranie listy publikacji .....   | 66 |
| 9.     | ApiMail .....   | 68 |
| 9.1.   | sendMail – wysyłka wiadomości e-mail .....  | 68 |
| 9.2.   | getMessages – pobieranie wiadomości .....   | 69 |
| 10.    | ApiUsers.....   | 71 |
| 10.1.  | getDeletedUsers – pobranie usuniętych użytkowników.....   | 71 |
| 11.    | ApiGuide .....  | 72 |
| 11.1.  | getDescription – pobranie opisu poradnika .....   | 72 |

|   |   |    |
|---|---|----|
| 11.2.                                   | startGuide – uruchomienie poradnika .....                             | 73 |
| 11.3.                                   | getQuestion – pobranie konkretnego pytania wraz z odpowiedziami ..... | 75 |
| 11.4.                                   | setAnswer – odpowiadanie na pytanie .....                             | 77 |
| 11.5.                                   | getSummary – pobranie podsumowania .....                              | 78 |
| 11.6.                                   | getPDFGuide – pobranie podsumowania wykonania poradnika .....         | 79 |
| 11.7.                                   | getList – Pobranie listy poradników .....                             | 80 |
| Historia wersji (zmian) dokumentu ..... |   | 83 |

## Wstęp

Do komunikacji z zewnętrznymi partnerami w projekcie PPK3 Baza Wiedzy wykorzystano wzorzec architektury REST, Representational State Transfer.

Parametry w zależności od metody przesyłane są metodą POST lub GET. Jeżeli dane są wysyłane metodą POST to w nagłówku wysyłanej wiadomości parametr Content-Type musi być ustawiony na „application/x-www-form-urlencoded”. Jest to standardowy format przesyłania parametrów metodą POST.

Wszystkie wyniki reprezentowane są przy użyciu zapisu JSON. Pełen opis tego formatu danych znajduje się w dokumencie [RFC 4627](http://tools.ietf.org/html/rfc4627) <http://tools.ietf.org/html/rfc4627>.

## Format odpowiedzi

Struktura każdej odpowiedzi wygląda następująco:

```
{
  "content" : < rezultat zapytania >,
  "language" : < kod tłumaczenia >,
  "status" : {
    "code" : < kod odpowiedzi/błędu >,
    "message" : < opis odpowiedzi/błędu >
  }
}
```

# Opis funkcji

## 1. ApiCommons

### 1.1. getVoivodeships - Słownik województwa

#### Opis

Metoda pobierająca słownik województw wraz z przyporządkowanym kodem teryt.

#### Nazwa funkcji

**getVoivodeships**

#### Parametry wejściowe

| Parametry wysyłane metodą POST |        |             |                  |                   |                                    |
|--------------------------------|--------|-------------|------------------|-------------------|------------------------------------|
| Nazwa zmiennej                 | Typ    | obowiązkowy | domyślna wartość | warunki           | opis                               |
| Lang                           | String | Nie         | pl               | poprawny kod lang | kod języka w standardzie ISO 639-1 |

#### Struktura danych wyjściowych

```
{
  "content" : [ {
    "Territory" : {
      "teryt" : < identyfikator terytu >,
      "name" : < nazwa województwa >
    }
  } ],
  "language" : < kod języka >,
  "status" : {
    "code" : < status_odpowiedzi >,
    "message" : < komunikat >
  }
}
```

#### Przykład wywołania

```
http://[adres_usługi]/ApiCommons/getVoivodeships
```

#### Opis kodów

```
OK_0000 poprawna odpowiedź
```

### 1.2. getPKD - Słownik PKD

#### Opis

Metoda pobierająca określony fragment drzewa PKD na podstawie przesłanego kodu PKD. W przypadku nie wysłania kodu PKD metoda zwraca główny kategorii PKD

#### Nazwa funkcji

**getPKD**

## Parametry wejściowe

| Parametry wysyłane metodą POST |        |             |                  |                   |                                    |
|--------------------------------|--------|-------------|------------------|-------------------|------------------------------------|
| Nazwa zmiennej                 | Typ    | obowiązkowy | domyślna wartość | warunki           | opis                               |
| code                           | string | nie         |                  | max 10 znaków     | kod PKD                            |
| lang                           | string | nie         | pl               | poprawny kod lang | kod języka w standardzie ISO 639-1 |

## Struktura danych wyjściowych

```
{
  "content" : {
    "PkdTreeFragment" : [ {
      "id" : < identyfikator PKD >,
      "code" : < kod PKD >,
      "title" : < nazwa PKD >,
      "hasChildren" : < informacja czy dany element posiada podgałęzie >,
      ===== WERSJA V2 =====
      "description" : < wyjaśnienia (obejmuje oraz nie obejmuje) >
      "lang" : < kod języka >
      ===== KONIEC WERSJI V2 =====
    } ]
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

## Przykład wywołania

```
http://[adres_usługi]/ApiCommons/getPkd
http://[adres_usługi]/ApiCommons/getPkd?code=A
http://[adres_usługi]/ApiCommons/getPkd?lang=en
http://[adres_usługi]/ApiCommons/getPkd?code=A&lang=en
```

## Opis kodów

OK\_0000 poprawna odpowiedź  
OK\_0002 poprawna odpowiedź brak wyników

ERR\_0006 Pole [nazwa pola] ma większą długość niż [max] znaków  
ERR\_0005 Pole lang posiada niepoprawny kod języka.

## 1.3. getPKDDetails - Pobranie szczegółów PKD

### Opis

Metoda pobierająca szczegóły PKD. W odpowiedzi dostajemy pełne informacje o PKD

### Nazwa funkcji

getPKDDetails

## Parametry wejściowe

| Parametry wysyłane metodą POST |        |             |                  |                   |                                    |
|--------------------------------|--------|-------------|------------------|-------------------|------------------------------------|
| Nazwa zmiennej                 | Typ    | obowiązkowy | domyślna wartość | warunki           | opis                               |
| id                             | int    | tak         |                  |                   | identyfikator PKD                  |
| lang                           | string | nie         | pl               | poprawny kod lang | kod języka w standardzie ISO 639-1 |

## Struktura danych wyjściowych

```
{
  "content" : {
    "PkdDetails" : {
      "id" : < identyfikator PKD >,
      "code" : < kod PKD >,
      "title" : < nazwa PKD >,
      "includes_description" : < podklasa obejmuje >,
      "not_includes_description" : < podklasa nie obejmuje >,
      ===== WERSJA V2 =====
      "for_natural_person" : < dla osób fizycznych >
      "related_procedures" : [ { < tablica powiązanych usług >
        "id" : < identyfikator usługi >,
        "name" : < nazwa usługi >,
        "alias" : < nazwa skrócona usługi >,
        "lang" : < język usługi >
      } ],
      "related_guides" [ { < tablica powiązanych poradników >
        "id" : < identyfikator poradnika >,
        "name" : < nazwa poradnika >,
        "alias" : < alias poradnika >,
        "lang" : < język poradnika >
      } ]
      ===== KONIEC WERSJI V2 =====
    }
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status_odpowiedzi >,
    "message" : < komunikat >
  }
}
```

## Przykład wywołania

```
http://[adres_uslugi]/ApiCommons/getPKDDetails?id=8242
http://[adres_uslugi]/ApiCommons/getPKDDetails?id=8242&lang=en
```

## Opis kodów

OK\_0000 poprawna odpowiedź

ERR\_0005 Pole lang posiada niepoprawny kod języka.

ERR\_0011 Pole [nazwa pola] jest wymagane.

ERR\_0014 Pole [nazwa pola] nie posiada prawidłowej wartości dla ID.

Content->for\_natural\_person:

- true
- false



## 1.4. getPKDList - wyszukiwanie PKD

Metoda wyszukuje PKD na podstawie kodu, opisu lub słów kluczowych (użyty mechanizm SPHINX)

### Nazwa funkcji

getPKDList

### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |                  |   |  |
|-------------------------------|--------|-------------|------------------|---|--|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślna wartość | warunki                                 | opis   |
| search                        | string | nie         |                  | max 255 znaków                          | wyszukiwana fraza – część słowa, pełna fraza lub pełen kod (z literą lub bez) podany w formacie scalonym lub z kropkami                                  |
| sort                          | string | nie         | weight           | weight, code, name, for_natural_person. | wartość z listy, przy czym „weight” sortuje dwustopniowo – po wadze, a potem po nazwie (jeśli nie ma search, waga jest stała, sortowanie jest po nazwie) |
| order                         | string | nie         | asc              | asc, desc                               | ustawia kierunek sortowanie listy po nazwie sytuacji życiowej, przyjmuje dwie możliwe wartości: <i>asc</i> lub <i>desc</i>                               |
| lang                          | string | nie         | pl               | poprawny kod lang                       | kod języka w standardzie ISO 639-1   |

### Struktura danych wyjściowych

```
{
  "content" : {
    "hint" : < fraza, która zastąpiła przypuszczalnie błędnie wpisane słowa >,
    "pkd_list" : [ {
      "id" : < wewnętrzny identyfikator PKD >,
      "code" : < kod PKD w formacie AA.BB.C >,
      "section" : < sekcja PKD >,
      "name" : < nazwa PKD >,
      "for_natural_person" : < dla osoby fizycznej: true/false >,
      "related_procedures" : < czy istnieją powiązane procedury: true/false >,
      "related_procedures_count" : < ilość powiązanych procedur >,
      "related_life_situations_count" : < ilość powiązanych poradników >,
      "weight" : < waga >,
    } ]
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

### Przykład wywołania

```
http://[adres_usługi]/ApiCommons/getPKDList
http://[adres_usługi]/ApiCommons/getPKDList?search=produkcja
```

## Opis kodów

OK\_0000 - poprawna odpowiedź  
OK\_0002 - poprawna odpowiedź, brak wyników

ERR\_0005 - Pole lang posiada niepoprawny kod języka.  
ERR\_0006 - Pole [nazwa pola] ma większą długość niż [max] znaków.  
ERR\_0024 - wartość pola z poza listy

## 1.5. getPKDHints – Pobranie podpowiedzi dla wyszukiwarki procedur po pkd

Metoda pobierająca podpowiedzi dla wyszukiwarki procedur po PKD na podstawie wpisanej frazy. Podpowiadają się pełne nazwy PKD

### Nazwa funkcji

getPKDHints

### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |                                  |  |
|-------------------------------|--------|-------------|-----------|----------------------------------|--|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki                          | opis   |
| search                        | string | tak         |           | max 100 znaków                   | początkowy fragment słowa znajdującego się w nazwach PKD |
| limit                         | int    | nie         | 10        | min wartość 1, max wartość 99999 | liczba wyników na stronie                                |
| lang                          | string | nie         | pl        | poprawny kod lang                | kod języka w standardzie ISO 639-1                       |

### Struktura danych wyjściowych

```
{
  "content" : [ {
    "name" : < nazwa PKD >
  } ],
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

### Przykład wywołania

[http://\[adres\\_usługi\]/ApiCommons/getPKDHints?search=przetwórstwo](http://[adres_usługi]/ApiCommons/getPKDHints?search=przetwórstwo)

### Opis kodów

OK\_0000 poprawna odpowiedź  
OK\_0002 poprawna odpowiedź nie odnaleziono wyników spełniających podane kryterium

ERR\_0004 - Niepoprawny typ pola  
ERR\_0005 - Pole lang posiada niepoprawny kod języka.  
ERR\_0006 - Pole [nazwa pola] ma większą długość niż [max] znaków.  
ERR\_0010 - Pole wykracza poza dopuszczalny zakres

## 1.6. getSYM – wyszukanie miejscowości w oparciu o tekst lub kod pocztowy

Metoda pozwala na wyszukanie prawidłowej miejscowości w oparciu o tekst wpisany przez użytkownika. Może być nim fragment nazwy miejscowości, kompletna nazwa miejscowości wraz z fragmentem gminy po przecinku, lub kompletny kod pocztowy.

System wyszukuje jedynie pełne miejscowości z bazy SIMC (bez miejscowości będących częściami innych miejscowości)

Wynikowo użytkownik dostaje listę miejscowości posortowaną wg preferencji miasto/wieś a następnie posortowaną alfabetycznie.

System wyszukuje nazwy wg następujących kryteriów:

- jeśli fraza wejściowa to kod pocztowy TRIM([0-9]{2}[ -]\*[0-9]{3}) system pyta o kod pocztowy, w innym wypadku o miejscowość
- system jest niezależny od ilości spacji i myślników w nazwach miejscowości i traktuje je zamiennie
- system zwraca uwagę na polskie znaki
- system nie zwraca uwagi na wielkość znaków
- jeśli we frazie jest przecinek, to rozdzielamy zapytanie na pytanie o miejscowość (przed przecinkiem) i pytanie o gminę (po przecinku)
- fraza przed przecinkiem - czyli miejscowość wyodrębniona z frazy podlega takim samym zasadom jak dla frazy tylko z samą miejscowością
- miejscowość nie może mieć mniej niż 2 znaków znaczących. Jeśli tak się stanie - system zgłosi wyjątek zbyt krótkiej frazy i zakończy proces
- pytanie o gminę zawsze jest nazwa\_gminy% niezależnie od liczby znaków
- jeśli fraza miejscowości <5 znaków a więcej niż 1 znak, wówczas zapytanie kierowane jest o dokładnie taką nazwę miejscowości jako podano
- jeśli fraza miejscowości >=5 znaków, to zapytanie kierowane jest do bazy o nazwa%
- jeśli nie ma odpowiedzi w wyniku zapytania, wówczas zanim odpowiemy użytkownikowi, system uruchamia zapytanie z eliminacją znaków diakrytycznych w tekście przy wyszukiwaniu, czyli zwróci np. „Poznań” po wpisaniu „Poznan”
- jeśli nadal nie ma odpowiedzi a fraza jest mniejsza niż 5 znaków to generowany jest wyjątek typu "brak wyników, podaj pełną lub dłuższą frazę"
- jeśli fraza miejscowości zawiera spację lub myślnik w środku (jest wielocłonowa) również generuj wyjątek typu "brak wyników"
- jeśli nadal nie ma odpowiedzi w wyniku zapytania to system korzysta z algorytmów wykrywających literówki w tekście wg zaszytego słownika i próbuje zadać pytanie bazę o sugerowany poprawny wynik
- jeśli i to nie pomogło - system zwraca wyjątek typu "brak wyników"

### Nazwa funkcji

getSYM

### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |  |  |
|-------------------------------|--------|-------------|-----------|--|--|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki  | opis   |
| search                        | string | tak         |           | Kod pocztowy w formacie xx-xxx lub ciąg tekstu długości 2-255 znaków | fragment nazwy miejscowości, kompletna nazwa miejscowości wraz z fragmentem gminy po przecinku, lub kompletny kod pocztowy |

## Struktura danych wyjściowych

```
{
  "content" :
  [{
    "localities" : <lista najbardziej pasujących miejscowości/gmin>
    [{
      "name" : <nazwa miejscowości wyświetlana użytkownikowi w standardzie: miejscowość
(gmina, województwo), jeśli SYM to miasto to nie wyświetla się gmina, tylko samo woj. >,
      "sym" : <kod SYM (baza SIMC) dla danej miejscowości np. 0879340>
    }],
    "rows_count" : < liczba rekordów ogółem >,
  }],
  "language" : < kod języka, zawsze PL >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >,
  }
}
```

## Przykłady wywołania

[http://\[adres\\_usługi\]/ApiCommons/getSYM?search=Warszawa](http://[adres_usługi]/ApiCommons/getSYM?search=Warszawa)

## Opis kodów

OK\_0000 poprawna odpowiedź

OK\_0002 poprawna odpowiedź nie odnaleziono wyników spełniających podane kryterium

OK\_0004 Nie znaleziono wyników. Wpisz pełną nazwę lub co najmniej 5 znaków.

ERR\_7002 Nie wprowadzono tekstu.

ERR\_7003 Nieprawidłowy kod pocztowy, wprowadź kod w formacie 'XX-XXX'.

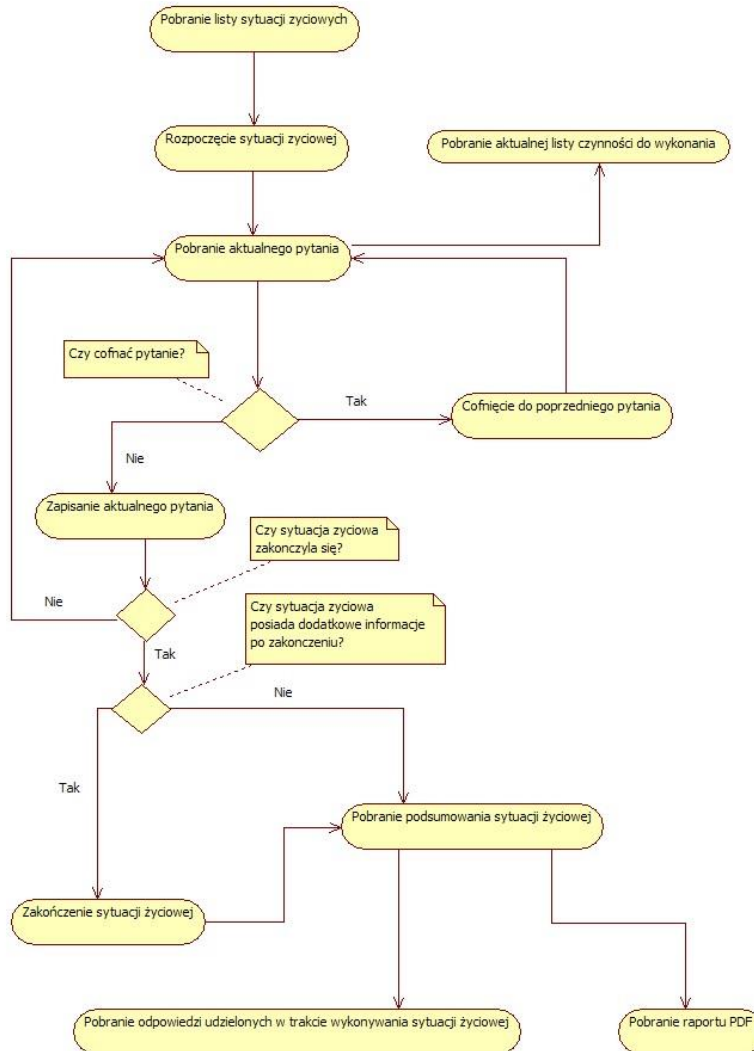
ERR\_7004 Nieprawidłowy format kodu pocztowego. Wprowadź kod w formacie liczbowym 'XX-XXX'.

ERR\_7005 Nazwa gminy nie powinna zawierać znaku ','.

## 2. ApiLifeSituations

### 2.1. Algorytm

Poniższy algorytm opisuje sposób poruszania się po metodach api sytuacji życiowych.



### 2.2. getPDF - Pobranie PDF-a

Metoda pobiera PDF-a dla ukończonej sytuacji życiowej

#### Nazwa funkcji

getPDF

#### Parametry wejściowe

| Parametry wysłane metodą POST |        |             |                  |         |   |
|-------------------------------|--------|-------------|------------------|---------|---|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślna wartość | warunki | opis                                      |
| hash                          | string | tak         |                  |         | identyfikator wykonanej sytuacji życiowej |

|         |     |     |   |                 |   |
|---------|-----|-----|---|-----------------|---|
| steps   | int | nie | 0 | wartość 0 lub 1 | flaga umożliwiająca wyświetlenie dokładnych informacji na temat szczegółów procedury (przebieg) |
| summary | int | nie | 0 | wartość 0 lub 1 | flaga umożliwiająca dodanie podsumowania oraz listę procedur do dokumentu PDF                   |

### Struktura danych wyjściowych

Jeżeli zostały poprawnie przekazane parametry wejściowe, po wykonaniu metody przekazany zostanie pdf

### Przykład wywołania

```
http://[adres_uslugi]/ApiLifeSituations/getPDF?hash=2164d0be48f216ce514ec76eda7f53c779c14ff4&steps=0
```

Opis kodów

```
ERR_0013 Wartość pola nie znajduje się na liście (0,1)
ERR_0004 Pole jest niepoprawnego typu
ERR_0011 Pole jest wymagane
ERR_5006 Sytuacja życiowa nie została ukończona.
```

## 2.3. startLifeSituation - start sytuacji życiowej

Metoda startuje sytuację życiową

### Nazwa funkcji

**startLifeSituation**

### Parametry wejściowe

| Parametry wysyłane metodą POST |        |             |                  |                         |  |
|--------------------------------|--------|-------------|------------------|-------------------------|--|
| Nazwa zmiennej                 | Typ    | obowiązkowy | domyślna wartość | warunki                 | opis   |
| acronym                        | string | tak         |                  | max 255 znaków          | Akronim, dla testowego wywołania może przyjąć postać *TEST-xxx1-xxx2*yyy gdzie xxx1 to id sytuacji życiowej xxx2 to timestamp yyy to akronim sytuacji życiowej w postaci „id-akronim” lub „id” lub „akronim” |
| lang                           | string | nie         | PL               | poprawny kod lang       | kod języka w standardzie ISO 639-1   |
| user_id                        | int    | nie         |                  | unikatowy identyfikator | identyfikator użytkownika  |

## Struktura danych wyjściowych

```
{
  "content" : {
    "description" : < opis startowy sytuacji życiowej >,
    "keywords" : < keywords >,
    "id" : < identyfikator sytuacji życiowej >,
    "name" : < nazwa sytuacji życiowej >,
    "category" : < kategoria >,
    "importance" : < ważność sytuacji życiowej z przedziału od 0 do 999 >,
    "icon" : < link do ikony sytuacji życiowej >,
    "smallDescription" : < skrócony opis >,
    "catalogues" : [ { < katalogi tematyczne powiązane z sytuacją życiową >
      "id" : < identyfikator katalogu >,
      "name" : < nazwa katalogu >
    } ],
    "language" : < język w którym może być opis >,
    "hash" : < hash >,
    "stages" : [ { < drzewo etapów >
      "stage_id" : < identyfikator etapu >,
      "process_id" : < identyfikator sytuacji życiowej związanej z etapem >,
      "title" : < tytuł etapu >,
      "substages" : [ { < drzewo podetapów >
        } ]
    } ]
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

## Przykład wywołania

```
http://[adres_usługi]/ApiLifeSituations/startLifeSituation?acronym=GABINET_PIELEGNIARSKI_R
http://[adres_usługi]/ApiLifeSituations/startLifeSituation?acronym=205-GABINET_PIELEGNIARSKI_R
http://[adres_usługi]/ApiLifeSituations/startLifeSituation?acronym=205
http://[adres_usługi]/ApiLifeSituations/startLifeSituation?acronym=*TEST-12-9999999*GABINET_PIELEGNIARSKI_R
```

## Opis kodów

OK\_0000 poprawna odpowiedź  
ERR\_0006 Pole acronym ma większą długość niż 255 znaków  
ERR\_0011 Pole jest wymagane  
ERR\_0005 Pole lang posiada niepoprawny kod języka.  
ERR\_5016 Jeden lub więcej podprocesów wchodzących w skład sytuacji życiowej nie jest opublikowany. Nie można rozpocząć sytuacji życiowej.  
ERR\_5018 - Pole [nazwa pola] posiada niepoprawną strukturę.  
ERR\_5019 - Przekroczono czas uruchomienia testowej sytuacji życiowej. Należy ponownie uruchomić podgląd sytuacji życiowej z zaplecza  
ERR\_5020 - Nieodnaleziono sytuacji życiowej o podanym ID

## 2.4. getCurrentQuestion - pobranie pytania

Metoda pobiera bieżące pytanie

### Nazwa funkcji

getCurrentQuestion

## Parametry wejściowe

| Parametry wysyłane metodą POST |        |             |                  |                   |                                    |
|--------------------------------|--------|-------------|------------------|-------------------|------------------------------------|
| Nazwa zmiennej                 | Typ    | obowiązkowy | Wartość domyślna | warunki           | opis                               |
| hash                           | string | tak         |                  | max 40 znaków     | hash                               |
| lang                           | string | nie         | PL               | poprawny kod lang | kod języka w standardzie ISO 639-1 |

## Struktura danych wyjściowych

```
{
  "content" : {
    "question" : {
      "question" : < pytanie >,
      "description" : < dodatkowy opis pytania >
      "answer_type" : <typ odpowiedzi, jednokrotnego lub wielokrotnego wyboru:
checkbox/radio>
    },
    "answers" : [ { < tablicaodpowiedzi >
      "answer_id" : < identyfikator odpowiedzi przekazywany do metody saveCurrentQuestion >,
      "answer" : < treść odpowiedzi >
      "is_exclusive": <dodatkowy parametr, który określa czy odpowiedź wyłącza możliwość
wyboru innych (może wystąpić 1 na dane pytanie o typie 'checkbox'): false/true>
    } ],
    "last_answer" : {
      "answer_id" : < ostatnia odpowiedz udzielona przez uzytkownika na to pytanie >
    },
    "actions" : [ { < tablica czynności >
      "url" : < adres URL >,
      "url_description" : < opis co znajduje się pod URL >,
      "description" : < opis czynności >,
      "title" : < nazwa czynności >,
      "type" : < typ czynności >
    } ],
    "stages" : [ { < drzewo etapów >
      "stage_id" : < identyfikator etapu >,
      "process_id" : < identyfikator sytuacji życiowej związanej z etapem >,
      "title" : < tytuł etapu >,
      "substages" : {
        } ]
    },
    "current_stage" : { < aktualny etap >
      "stage_id" : < identyfikator etapu >,
      "process_id" : < identyfikator procesu >
    }
  },
  "language" : pl,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

## Przykład wywołania

```
http://[adres_uslugi]/ApilifeSituations/getCurrentQuestion?hash=2164d0be48f216ce514ec76eda7f53c779c14ff4
```

## Opis kodów

```
OK_0000 Poprawna odpowiedź
OK_0001 Zakończono wykonywanie sytuacji życiowej
```



ERR\_0011 Pole hash jest wymagane  
ERR\_0005 Pole lang posiada niepoprawny kod języka  
ERR\_0022 Pole hash posiada nieprawidłową wartość hash.  
ERR\_5008 Brak sesji o podanym identyfikatorze.  
ERR\_5009 Sytuacja życiowa posiada nieprawidłową strukturę.

## 2.5. saveCurrentQuestion – zapisanie bieżącego pytania

Metoda zapisuje odpowiedź na bieżące pytanie

### Nazwa funkcji

**saveCurrentQuestion**

### Parametry wejściowe

| Parametry wysyłane metodą POST |        |             |                |                          |
|--------------------------------|--------|-------------|----------------|--------------------------|
| Nazwa zmiennej                 | Typ    | obowiązkowy | warunki        | opis                     |
| hash                           | string | tak         | max 40 znaków  | hash                     |
| answer_id                      | int    | tak         | weryfikacji id | identyfikator odpowiedzi |

### Struktura danych wyjściowych

```
{
  content : {
    status : < status >
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status_odpowiedzi >,
    "message" : < komunikat >
  }
}
```

### Przykład wywołania

[http://\[adres\\_usługi\]/saveCurrentQuestion?hash=2164d0be48f216ce514ec76eda7f53c779c14ff4](http://[adres_usługi]/saveCurrentQuestion?hash=2164d0be48f216ce514ec76eda7f53c779c14ff4)

Opis kodów

OK\_0000 poprawna odpowiedź  
ERR\_0004 Pole answer\_id posiada niepoprawny typ. Wymagany typ to INT  
ERR\_0011 Pole jest wymagane  
ERR\_0022 Pole hash posiada nieprawidłową wartość hash  
ERR\_5003 Brak odpowiedzi dla aktualnego stanu sesji  
ERR\_5004 Taka odpowiedź nie istnieje dla tego pytania  
ERR\_5024 Dla pytania typu RADIO odpowiedź musi przyjmować jedną wartość  
ERR\_5023 Zaznaczono w odpowiedzi wiele wartości, spośród których jedna posiada atrybut EXCLUSIVE

## 2.6. revertToPreviousQuestion - cofnięcie do poprzedniego pytania

Metoda cofa do poprzedniego pytania

### Nazwa funkcji

**revertToPreviousQuestion**

## Parametry wejściowe

| Parametry wysyłane metodą POST |        |             |               |      |
|--------------------------------|--------|-------------|---------------|------|
| Nazwa zmiennej                 | Typ    | obowiązkowy | warunki       | opis |
| hash                           | string | tak         | max 40 znaków | hash |

## Struktura danych wyjściowych

```
{
  "content" : {
    "status" : < status >
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

## Przykład wywołania

[http://\[adres\\_usługi\]/ApiLifeSituations/revertToPreviousQuestion?hash=2164d0be48f216ce514ec76e da7f53c779c14ff4](http://[adres_usługi]/ApiLifeSituations/revertToPreviousQuestion?hash=2164d0be48f216ce514ec76e da7f53c779c14ff4)

## Opis kodów

OK\_0000 poprawna odpowiedź

ERR\_0011 Pole jest wymagane

ERR\_5007 Sytuacja życiowa została zakończona. Nie ma możliwości cofnięcia się

ERR\_5001 Wystąpił błąd podczas wycofywania odpowiedzi

## 2.7. revertToStage – powrót do wybranego etapu

Metoda umożliwia powrót do pierwszego pytania wybranego etapu

### Nazwa funkcji

**revertToStage**

### Parametry wejściowe

| Parametry wysyłane metodą POST |        |             |               |                                 |
|--------------------------------|--------|-------------|---------------|---------------------------------|
| Nazwa zmiennej                 | Typ    | obowiązkowy | warunki       | opis                            |
| hash                           | string | tak         | max 40 znaków | Hash                            |
| stage_id                       | Int    | Tak         |               | Identyfikator etapu             |
| process_id                     | Int    | Tak         |               | Identyfikator sytuacji życiowej |

## Struktura danych wyjściowych

```
{
  "content" : {
    "status" : < status >
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

```
}
```

### Przykład wywołania

```
http://[adres_uslugi]/ApiLifeSituations/revertToStage?hash=af1ec08554448b55d2a1b2757b725da89c672f84&process_id=1244&stage_id=0
```

### Opis kodów

OK\_0000 poprawna odpowiedź

ERR\_0011 Pole jest wymagane

ERR\_5007 Sytuacja życiowa została zakończona. Nie ma możliwości cofnięcia się

ERR\_5001 Wystąpił błąd podczas wycofywania odpowiedzi

ERR\_5008 Sesja nie istnieje

ERR\_5012 Taka sytuacja życiowa nie istnieje

ERR\_5013 Taki etap dla podanej sytuacji życiowej nie istnieje

ERR\_5014 W tej sesji nie występuje sytuacja życiowa o podanej strukturze

ERR\_5015 Nie można cofnąć się do etapu który nie został jeszcze wykonany

## 2.8. getEndDescription - pobranie końcowego opisu

Metoda pobiera opis zakończonej sytuacji życiowej

### Nazwa funkcji

**getEndDescription**

### Parametry wejściowe

| Parametry wysyłane metodą POST |        |             |               |      |
|--------------------------------|--------|-------------|---------------|------|
| Nazwa zmiennej                 | Typ    | obowiązkowy | warunki       | opis |
| hash                           | string | tak         | max 40 znaków | hash |

### Struktura danych wyjściowych

```
{
  "content" : {
    "title" : < tytuł opisu >,
    "description" : < tresc opisu >
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

### Przykład wywołania

```
http://[adres_uslugi]/ApiLifeSituations/getEndDescription?hash=2164d0be48f216ce514ec76eda7f53c779c14ff4
```

### Opis kodów

OK\_0000 poprawna odpowiedź

ERR\_0011 Pole jest wymagany

ERR\_5006 Sytuacja życiowa nie została zakończona

## 2.9. getSummary - pobranie podsumowania

Metoda pobiera podsumowanie sytuacji życiowej

### Nazwa funkcji

**getSummary**

### Parametry wejściowe

| Parametry wysyłane metodą POST |        |             |               |      |
|--------------------------------|--------|-------------|---------------|------|
| Nazwa zmiennej                 | Typ    | obowiązkowy | warunki       | opis |
| hash                           | string | tak         | max 40 znaków | hash |

### Struktura danych wyjściowych

```
{
  "content" : { < lista czynności do wykonania >
    "url" : < odnośnik do artykułu lub procedury >,
    "url_description" : < opis odnośnika >,
    "description" : < opis czynności >,
    "title" : < tytuł/nazwa czynności >,
    "type" : < typ czynności >
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

### Przykład wywołania

```
http://[adres_usługi]/ApiLifeSituations/getSummary?hash=2164d0be48f216ce514ec76eda7f53c779c14ff4
```

### Opis kodów

OK\_0000 poprawna odpowiedź

ERR\_0011 Pole jest wymagany

ERR\_5006 Sytuacja życiowa nie została zakończona

## 2.10. getAnswers - Metoda pobiera odpowiedzi sytuacji życiowej

Metoda zwraca listę wszystkich pytań wraz z udzielonymi odpowiedziami

### Nazwa funkcji

**getAnswers**

### Parametry wejściowe

| Parametry wysyłane metodą POST |        |             |               |      |
|--------------------------------|--------|-------------|---------------|------|
| Nazwa zmiennej                 | Typ    | obowiązkowy | warunki       | opis |
| hash                           | string | tak         | max 40 znaków | hash |

### Struktura danych wyjściowych

```
{
```

```

"content" : { < lista udzielonych odpowiedzi >
  {
    "question" : < pytanie >,
    "answer" : < udzielona odpowiedź >
  }
},
"language" : < kod języka >,
"status" : {
  "code" : < status odpowiedzi >,
  "message" : < komunikat >
}
}

```

### Przykład wywołania

```

http://[adres_usługi]/ApiLifeSituations/getAnswers?hash=2164d0be48f216ce514ec76eda7f53c779c14ff4

```

### Opis kodów

OK\_0000 poprawna odpowiedź

ERR\_0011 Pole jest wymagany

ERR\_5006 Sytuacja życiowa nie została zakończona

## 2.11. getList – Pobranie listy sytuacji życiowych

Metoda zwraca listę wszystkich sytuacji życiowych posortowanych po nazwie sytuacji życiowej. Możliwe jest podanie parametru **search**, **lang**, **limit**, **page**, **sort**. W przypadku pobierania listy w języku innym niż język polski, jeśli nie została znaleziona sytuacja życiowa w żądany języku zwracana jest sytuacja życiowa w języku polskim

### Nazwa funkcji

**getList**

### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |                                    |  |
|-------------------------------|--------|-------------|-----------|------------------------------------|--|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki                            | opis   |
| search                        | string | nie         |           | max 100 znaków                     | umożliwia zawężenie zwracanej listy do elementów wyszukanych po nazwie lub po słowach kluczowych na podstawie przekazanego parametru |
| lang                          | string | nie         | PL        | poprawny kod lang                  | kod języka w standardzie ISO 639-1   |
| limit                         | int    | nie         | 10        | max wartość 9999                   | ustawia ilość zwracanych jednorazowo elementów listy   |
| page                          | int    | nie         | 1         | max wartość 999                    | ustawia numer strony która ma zostać zwrócona, współpracuje z parametrem limit   |
| order                         | string | nie         | asc       | na liście (asc, desc), max 4 znaki | ustawia kierunek sortowanie listy po nazwie sytuacji życiowej, przyjmuje dwie możliwe wartości: <i>asc</i> lub <i>desc</i>           |

|            |        |     |      |   |   |
|------------|--------|-----|------|---|---|
| category   | array  | nie |      | Wartość z listy musi być zgodna ze słownikiem | Lista kategorii podana po przecinku z zakresu :<br>'ZAK' => 'Zakładam firmę',<br>'PRO' => 'Prowadzę firmę',<br>'ZAM' => 'Zamykam firmę',<br>'USL' => 'Tymczasowo świadczę usługi na terenie Polski' |
| importance | Int    | Nie |      | Z zakresu od 0 do 999                         | Określa ważność z przedziału od 0 do 999  |
| sort       | String | Nie | Name | Na liście, max 25 znaków                      | Wartość z listy name, category,importance   |
| pkd        | string | nie |      |   | Poprawny kod PKD, z kropkami lub bez  |

### Struktura danych wyjściowych

```
{
  "content" : {
    "rows_count" : < liczba wszystkich wierszy >,
    "LifeSituations" : [ {
      "LifeSituation" : {
        "id" : < identyfikator sytuacji życiowej >,
        "acronym" : < akronim sytuacji życiowej >,
        "category" : < kod kategorii >,
        "importance" : < ważność poradnika >,
        =====WERSJA V4=====
        "weight" : < waga >,
        =====KONIEC WERSJI V4=====
        "icon" : < link do ikonki sytuacji życiowych >,
        "name" : < nazwa sytuacji życiowej >,
        "language" : < język w którym został zwrócony element listy >,
        =====WERSJA V2=====
        "short_introduction" : < zawartość pola „Krótkie wprowadzenie” >
        =====KONIEC WERSJI V2=====
        =====WERSJA V3=====
        "modification_date" : < data modyfikacji >,
        =====KONIEC WERSJI V3=====
      }
    } ]
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

### Przykład wywołania

```
http://[adres_usługi]/ApiLifeSituations/getList
http://[adres_usługi]/ApiLifeSituations/getList?limit=500
http://[adres_usługi]/ApiLifeSituations/getList?limit=500&search=otwieram
```

### Opis kodów

```
OK_0000 - poprawna odpowiedź
ERR_0004 - pole posiada [nazwa_pola] niepoprawny typ. Wymagany typ to [typ]
ERR_0005 - pole [nazwa_pola] posiada niepoprawny kod języka
ERR_0006 - pole [nazwa_pola] ma większą długość niż [wymagana_długość] znaków.
```

ERR\_0007 - pole [nazwa\_pola] większą wartość niż [maksymalna wartość].  
 ERR\_0007 - pole [nazwa\_pola] mniejsza wartość niż [maksymalna wartość].  
 ERR\_0013 - pole [nazwa\_pola] nie znajduje się na liście [wymagana lista parametrów].  
 ERR\_0024 - wartość pola z poza listy  
 ERR\_0025 - Wartość pola pkd jest niepoprawnym kodem PKD  
 ERR\_0035 - Żądana strona wykracza poza listę wyników

## 2.12. getDescription Pobranie opisu sytuacji życiowej

Metoda zwraca opis sytuacji życiowej. Wymagane jest podanie parametru **acronym**, możliwe jest podanie parametru **lang**. Wartość pola **acronym** to złączenie „id-akronim” lub „id” lub „akronim”. W przypadku pobierania opisu w języku innym niż język polski, jeśli nie został znaleziony opis sytuacji życiowej w żądany języku zwracany jest opis sytuacji życiowej w języku polskim

### Nazwa funkcji

getDescription

### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |                   |   |
|-------------------------------|--------|-------------|-----------|-------------------|---|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki           | opis  |
| acronym                       | string | tak         |           | max 255 znaków    | akronim sytuacji życiowej, otrzymany po wywołaniu funkcji getList<br>Uwaga: akronim może przyjmować również format:<br>*TEST-[id sytuacji życiowej]-[timestamp]*[Akronim sytuacji życiowej] |
| lang                          | string | nie         | PL        | poprawny kod lang | kod języka w standardzie ISO 639-1  |

### Struktura danych wyjściowych

```
{
  "content" : {
    "LifeSituationAction" : {
      "acronym" : < akronim sytuacji życiowej >,
      "description" : < zwrócony opis sytuacji życiowej >,
      "keywords" : < słowa kluczowe >,
      "smallDescription" : < skrócony opis >,
      "id" : < identyfikator sytuacji życiowej >,
      "name" : < nazwa sytuacji życiowej >,
      "category" : < kod kategorii sytuacji życiowej >,
      "importance" : < ważność sytuacji życiowej z przedziału od 0 do 999 >,
      "icon" : < link do ikony >,
      "description" : < opis sytuacji życiowej >,
      "catalogues" : [ { < katalogi tematyczne powiązane z sytuacją życiową >
        "id" : < identyfikator katalogu >,
        "name" : < nazwa katalogu >
      } ],
      "relatedPublications" : [ { < publikacje powiązane z sytuacją życiową >
        "alias" : < alias publikacji >,
        "title" : < tytuł publikacji >,
        "language" : < język, w którym zwrócony został tytuł publikacji >
      } ],
      "acts" : [ { < tablica aktów prawnych powiązanych z poradnikiem >
        "id" : < id aktu prawnego >,
        "name" : < nazwa aktu prawnego >,
        "link" : < link do aktu prawnego >,
        "type_code" : < kod aktu prawnego Us; Ro; Uch; Pu; In >,
      } ],
    }
  }
}
```

```

        "type_description" : < opis kodu aktu prawnego (Ustawa; Rozporządzenie; Uchwała;
Przepisy unijne; Inny >,
        "language" : < język, w którym zwrócony został tytuł aktu prawnego >,
        "references" : [ { < lista nawiązań do innych aktów prawnych >
            "id" : < id aktu prawnego >,
            "name" : < nazwa aktu prawnego >,
            "link" : < link do aktu prawnego >,
            "type_code" : < kod aktu prawnego Us; Ro; Uch; Pu; In >,
            "type_description" : < opis kodu aktu prawnego (Ustawa; Rozporządzenie; Uchwała;
Przepisy unijne; Inny >,
            "language" : < język, w którym zwrócony został tytuł nawiązania do aktu prawnego
>,
        } ]
    } ],
    "pkd" : {
        "status_code" : < kod statusu PKD (NIE_DOTYCZY; WYBRANE >,
        "status_description" : < opis statusu pobrany ze słownika (np. Poradnik dotyczy
wybranych gałęzi PKD) >,
        "list" : [ { < tablica obiektów PKD >
            "code" : < kod PKD >,
            "title" : < nazwa gałęzi PKD >,
            "includes_description" : < opis informujący czego dotyczy PKD >,
            "not_includes_description" : < opis informujący czego nie dotyczy PKD >
        } ],
    },
    "language" : < kod języka w którym została zwrócona odpowiedź >
=====WERSJA V2=====
        "visible_acts" : [ { < tablica aktów prawnych powiązanych z poradnikiem >
            "id" : < id aktu prawnego >,
            "name" : < nazwa aktu prawnego >,
            "link" : < link do aktu prawnego >,
            "type_code" : < kod aktu prawnego Us; Ro; Uch; Pu; In >,
            "type_description" : < opis kodu aktu prawnego (Ustawa; Rozporządzenie; Uchwała;
Przepisy unijne; Inny >,
            "language" : < język, w którym zwrócony został tytuł aktu prawnego >,
            "references" : [ { < lista nawiązań do innych aktów prawnych >
                "id" : < id aktu prawnego >,
                "name" : < nazwa aktu prawnego >,
                "link" : < link do aktu prawnego >,
                "type_code" : < kod aktu prawnego Us; Ro; Uch; Pu; In >,
                "type_description" : < opis kodu aktu prawnego (Ustawa; Rozporządzenie;
Uchwała; Przepisy unijne; Inny >,
                "language" : < język, w którym zwrócony został tytuł nawiązania do aktu
prawnego >,
            } ]
        } ],
=====KONIEC WERSJI V2=====

    }
},
"language" : < kod języka >,
"status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
}
}

```

### Przykład wywołania

```

http://[adres_usługi]/ApiLifeSituations/getDescription
http://[adres_usługi]/ApiLifeSituations/getDescription?acronym=GABINET_PIELEGNIARSKI_R
http://[adres_usługi]/ApiLifeSituations/getDescription?acronym=205-GABINET_PIELEGNIARSKI_R
http://[adres_usługi]/ApiLifeSituations/getDescription?acronym=205
http://[adres_usługi]/ApiLifeSituations/getDescription?acronym=GABINET_PIELEGNIARSKI_R&lang=en

```



## Opis kodów

OK\_0000 poprawna odpowiedź  
ERR\_0011 - pole [nazwa\_pola] jest wymagane.  
ERR\_0005 - pole [nazwa\_pola] posiada niepoprawny kod języka  
ERR\_0006 - pole [nazwa\_pola] ma większą długość niż [wymagana\_długość] znaków  
ERR\_0100 - znaleziono więcej niż jedną sytuację życiową o podanym akronimie  
ERR\_5012 - Taka sytuacja życiowa nie istnieje.  
ERR\_5017 - Brak opisu dla podanej sytuacji życiowej  
ERR\_5018 - niepoprawna struktura akronimu(zawiera niepoprawne znaki)  
ERR\_5019 - Przekroczono czas uruchomienia testowej sytuacji życiowej. Należy ponownie uruchomić podgląd sytuacji życiowej z zaplecza  
ERR\_5020 - Nieodnaleziono sytuacji życiowej o podanym ID

Dodano w wersji V\_1.0.41:

"acronym" : < akronim sytuacji życiowej > ,

## 2.13. linkSessionWithUser Powiązanie sesji sytuacji życiowej z użytkownikiem

Metoda umożliwia połączenie sytuacji życiowej z identyfikatorem użytkownika. Metoda wywoływana w POST.

### Nazwa funkcji

**linkSessionWithUser**

### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |                |                           |
|-------------------------------|--------|-------------|-----------|----------------|---------------------------|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki        | opis                      |
| hash                          | string | tak         |           | max 255 znaków | hash                      |
| user_id                       | int    | tak         |           |                | Identyfikator użytkownika |

### Struktura danych wyjściowych

```
{
  "content" : {
    "status" : "OK"
  },
  "language" : < kod języka > ,
  "status" : {
    "code" : < status odpowiedzi > ,
    "message" : < komunikat >
  }
}
```

### Przykład wywołania

http://[adres\_usługi]/ApiLifeSituations/linkSessionWithUser

## Opis kodów

OK\_0000 poprawna odpowiedź  
ERR\_5011 - Sesja o podanym identyfikatorze jest już powiązana z użytkownikiem

## 2.14. getVisibleLifeSituationsPropositions Pobranie listy widocznych propozycji sytuacji życiowych

Metoda listę wszystkich propozycji sytuacji życiowych o statusie „WIDOCZNY”. Możliwe jest podanie parametru **lang** (domyślnie *pl*), aby zwrócić listę w innym niż domyślnie języku (TODO).

### Nazwa funkcji

**getVisibleLifeSituationsPropositions**

## Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |                   |                                    |
|-------------------------------|--------|-------------|-----------|-------------------|------------------------------------|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki           | opis                               |
| lang                          | string | nie         | PL        | poprawny kod lang | kod języka w standardzie ISO 639-1 |

## Struktura danych wyjściowych

```
[ {
  "LifeSituationsProposition" : {
    "szp_id" : < status propozycji >,
    "szp_status" : < status propozycji >,
    "szp_liczba_glosow" : < liczba oddanych głosów na propozycję >,
    "szp_temat" : < nazwa propozycji >,
    "szp_widoczny_od" : < data rozpoczęcia wyświetlania >,
    "szp_data_utworzenia" : < data utworzenia >,
    "szp_utworzyl" : < nazwa użytkownika tworzącego propozycję >,
    "szp_data_modyfikacji" : < data modyfikacji propozycji >,
    "szp_modyfikowal" : < nazwa użytkownika modyfikującego >,
    "szp_widoczny_do" : < data graniczna widoczności propozycji >,
    "szp_utworzyl_id" : < identyfikator użytkownika tworzącego propozycję >,
    "szp_modyfikowal_id" : < identyfikator użytkownika modyfikującego propozycję >,
    "id" : < identyfikator propozycji >,
    "szp_widoczny_dni" : < liczba >,
    "created" : < data utworzenia >,
    "created_by" : < osoba która utworzyła propozycję >,
    "modified" : < data modyfikacji >,
    "modified_by" : < osoba która zmodyfikowała propozycję >
  }
} ]
```

## Przykład wywołania

[http://\[adres\\_uslugi\]/ApiLifeSituations/getVisibleLifeSituationsPropositions](http://[adres_uslugi]/ApiLifeSituations/getVisibleLifeSituationsPropositions)

## Opis kodów

OK\_0000 poprawna odpowiedź  
ERR\_0005 Parametr „lang” niezgodny ze standardem ISO 639-1

## 2.15. voteOnLifeSituationProposition Głosowanie na propozycję sytuacji życiowej

Metoda umożliwiająca oddanie głosu na wybraną propozycję sytuacji życiowej lub dodanie własnej propozycji sytuacji życiowej.

## Nazwa funkcji

**voteOnLifeSituationProposition**

## Parametry wejściowe

| Parametry wysyłane metodą POST |        |             |           |                 |   |
|--------------------------------|--------|-------------|-----------|-----------------|---|
| Nazwa zmiennej                 | Typ    | obowiązkowy | domyślnie | warunki         | opis  |
| id                             | int    | nie         |           |                 | identyfikator propozycji sytuacji życiowej    |
| text                           | string | nie         |           | max 2000 znaków | temat propozycji sytuacji życiowej            |
| user                           | string | nie         |           | max 150 znaków  | imię i nazwisko osoby zgłaszającej propozycję |

## Struktura danych wyjściowych

```
{
  content : {
    voteStatus : < status głosu >
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status_odpowiedzi >,
    "message" : < komunikat >
  }
}
```

## Przykład wywołania

### Głosowanie na istniejącą propozycję

```
http://[adres_usługi]/ApiLifeSituations/voteOnLifeSituationProposition
W POST
id=12
```

### Dodanie własnej propozycji

```
http://[adres_usługi]/ApiLifeSituations/voteOnLifeSituationProposition
W POST
text=To jest moja propozycja
```

### Dodanie własnej propozycji przez użytkownika zalogowanego

```
http://[adres_usługi]/ApiLifeSituations/voteOnLifeSituationProposition
W POST
text=Moja propozycja&user=Jan Kowalski
```

## Opis kodów

OK\_0000 poprawna odpowiedź

ERR\_0004 Pole posiada niepoprawny typ. Wymagany typ to

ERR\_0006 Pole ma większą długość niż znaków

ERR\_2001 Brak propozycji o podanym ID.

ERR\_2002 Nie udało się zaaktualizować licznika głosów dla podanej propozycji sytuacji życiowej.

ERR\_2003 Nie podano parametrów wejściowych

ERR\_0001 Dane należy wysłać metodą POST

## 3. ApiOpinions

### 3.1. saveOpinionForContent - Dodawanie opinii

Metoda umożliwiająca dodanie opinii odnośnie artykułu, sytuacji życiowej, publikacji, procedury

#### Nazwa funkcji

saveOpinionForContent

#### Parametry wejściowe

| Parametry wysyłane metodą POST |        |                          |           |  |  |
|--------------------------------|--------|--------------------------|-----------|--|--|
| Nazwa zmiennej                 | Typ    | obowiązkowy              | domyślnie | warunki  | opis   |
| rate                           | string | tak                      |           | na liście (TAK, NIE, TROCHE)   | wartość oceny                                    |
| comment                        | string | nie                      |           | max 5000 znaków  | komentarz do oceny                               |
| group                          | string | tak                      |           | na liście (ARTYKULY, PUBLIKACJE, SYTUACJE, PROCEDURY, INSTYTUCJE, INNE)            | kod określający czego dotyczy opinia             |
| context_id                     | int    | tak, jeśli group != INNE |           |  | identyfikator ocenianego elementu                |
| context_title                  | string | tak, jeśli group == INNE |           | max 255 znaków; jeśli group != {INNE, ARTYKULY}, to parametr nie jest uwzględniany | tytuł ocenianego elementu                        |
| context_link                   | string | tak, jeśli group == INNE |           | max 255 znaków; jeśli group != {INNE, ARTYKULY}, to parametr nie jest uwzględniany | link do ocenianego elementu (URL absolutny)      |
| user_id                        | int    | nie                      |           |  | identyfikator użytkownika, który utworzył opinię |
| user_ip                        | string | tak                      |           | max 255 znaków   | adres IP użytkownika dodającego opinię           |

#### Struktura danych wyjściowych

```
{
  content : {
    voteStatus : < status głosu >,
    voteCount : < liczba głosów >
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status_odpowiedzi >,
    "message" : < komunikat >
  }
}
```

#### Przykład wywołania

##### Wysłanie opinii z komentarzem

`http://[adres_uslugi]/ApiOpinions/saveOpinionForContent`

W POST

`rate=TAK&comment=[Tresc komentarza]&group=ARTYKULY&context_id=1&context_title=Tytul  
artykulu&context_link=http://eu-go.gov.pl/artykuly/1&user_ip=234.231.121.2`

Wysłanie opinii bez komentarza

`http://[adres_uslugi]/ApiOpinions/saveOpinionForContent`

W POST rate=TAK&group=ARTYKULY&context\_id=1&context\_title=Tytul  
artykulu&context\_link=http://eu-go.gov.pl/artykuly/1&user\_ip=234.231.121.2

### Opis kodów

OK\_0000 poprawna odpowiedź

ERR\_0001 Dane należy wysłać metodą POST

ERR\_3001 Nie udało się dodać nowej opinii

ERR\_0004 - Niepoprawny typ pola

ERR\_0006 - Przekroczona dopuszczalna długość pola

ERR\_0010 - Pole wykracza poza dopuszczalny zakres

ERR\_0011 - Brak pola obowiązkowego

ERR\_0013 - Wartość nie znajduje się na liście

ERR\_0021 - Wartość pola jest niepoprawnym adresem IP

## 4. ApiInstitutions

### 4.1. saveInstitutionProposition - Zapisywanie propozycji nowej instytucji

Metoda umożliwiająca wysłanie przez użytkownika portalu propozycji nowej instytucji

#### Nazwa funkcji

**saveInstitutionProposition**

#### Parametry wejściowe

| Parametry wysyłane metodą POST |        |                           |           |  |  |
|--------------------------------|--------|---------------------------|-----------|--|--|
| Nazwa zmiennej                 | Typ    | obowiązkowy               | domyślnie | warunki                                      | opis                                   |
| name                           | string | tak                       |           | max 255 znaków                               | nazwa instytucji                       |
| type_id                        | int    | tak                       |           |  | id typu instytucji                     |
| nip                            | string | nie                       |           | walidacja NIP                                | NIP                                    |
| regon                          | string | tak                       |           | walidacja REGON                              | REGON                                  |
| street                         | string | tak                       |           | max 100 znaków                               | ulica                                  |
| number                         | string | nie                       |           | max 50 znaków                                | numer                                  |
| post_code                      | string | tak                       |           | walidacja kodu pocztowego                    | kod pocztowy                           |
| city                           | string | tak                       |           | max 100 znaków                               | mięscowość                             |
| voivodeship                    | string | tak                       |           | walidacja id województwa                     | województwo                            |
| longitude                      | string | nie                       |           | walidacja współrzędnych                      | długość geograficzna                   |
| latitude                       | string | nie                       |           | walidacja współrzędnych                      | szerokość geograficzna                 |
| email                          | string | tak                       |           | walidacja email, max 256 znaków              | email kontaktowy                       |
| www                            | string | tak (jeśli brak pola bip) |           | walidacja www, max 256 znaków                | strona www                             |
| bip                            | string | tak (jeśli brak pola www) |           | walidacja www, max 256 znaków                | strona bip                             |
| logo                           | string | nie                       |           | tekst base64, max długość 5*1024*1024 znaków | obraz zakodowany przy użyciu base64    |
| logo_ext                       | string | nie                       |           | max 256 znaków                               | rozszerzenie pliku logo                |
| activities                     | string | nie                       |           | max 256 znaków                               | zakres działalności                    |
| reported_id                    | int    | tak                       |           |  | id użytkownika wysyłającego zgłoszenie |

|                |        |     |  |  |   |
|----------------|--------|-----|--|--|---|
| reported_name  | string | nie |  | max 100 znaków                           | imię i nazwisko użytkownika wysyłającego zgłoszenie |
| reported_email | string | nie |  | walidacja email, max 256 znaków          | e-mail użytkownika wysyłającego zgłoszenie          |
| reported_ip    | string | nie |  | walidacja adresu ip                      | adres IP użytkownika wysyłającego zgłoszenie        |
| workingHours   | string | nie |  | max 256 znaków                           | godziny otwarcia                                    |
| phone          | string | nie |  | walidacja numeru telefonu, max 50 znaków | telefon kontaktowy                                  |
| fax            | string | nie |  | walidacja numeru telefonu, max 50 znaków | fax   |
| other_info     | string | nie |  | max 256 znaków                           | inne informacje                                     |

### Struktura danych wyjściowych

```
{
  "content" : {
    "institutionPropositionSaveStatus" : < status >
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

### Przykład wywołania

[http://\[adres\\_usługi\]/ApiInstitutions/saveInstitutionProposition](http://[adres_usługi]/ApiInstitutions/saveInstitutionProposition)

W POST:

```
name=Przykładowy Urząd&type_id=74&nip=9717789134&regon=950056002&street=Powstańców
Warszawskich&number=5&
post_code=60-
343&city=Poznań&voivodeship=30&longitude=16.94091796875&latitude=52.429222277955134&www=http:/
/przyklad.www.pl&bip=http://przyklad.bip.pl&logo=[logo_base64]&logo_ext=png
activities=zakres działalności&reported_id=10485&reported_name=Jan
Kowalski&reported_email=jan@kowalski.pl&
reported_ip=43.16.1.76&email=przyklad@email.pl&workingHours=>6-
16&phone=554434442&fax=111111111&
other_info=inne
```

### Opis kodów

OK\_0000 poprawna odpowiedź

ERR\_0001 Dane należy wysłać metodą POST  
 ERR\_0002 Pole %s jest niepoprawnym ciągiem base64.  
 ERR\_0003 Pole %s ma niepoprawny adres e-mail.  
 ERR\_0004 Pole %s posiada niepoprawny typ. Wymagany typ to %s.  
 ERR\_0006 Pole %s ma większą długość niż %s znaków.  
 ERR\_0009 Pole %s posiada niepoprawny numer telefonu.  
 ERR\_0011 Pole %s jest wymagane.  
 ERR\_0012 Pole %s posiada niepoprawny adres URL.

ERR\_0013 Pole %s nie znajduje się na liście (%s).  
 ERR\_0014 Pole %s nie posiada prawidłowej wartości dla ID.  
 ERR\_0015 Pole %s posiada nieprawidłowe znaki  
 ERR\_0016 Wartość pola %s jest niepoprawnym ciągiem NIP  
 ERR\_0017 Wartość pola %s jest niepoprawnym ciągiem REGON  
 ERR\_0018 Pole %s posiada nieprawidłową sumę kontrolną  
 ERR\_0019 Wartość pola %s jest niepoprawnym ciągiem kodu pocztowego  
 ERR\_0020 Wartość pola %s jest niepoprawnym koordynatem Latitude lub Longitude  
 ERR\_0021 Wartość pola %s jest niepoprawnym adresem IP  
 ERR\_4002 Nie udało się dodać nowej propozycji instytucji  
 ERR\_4006 Brak parametru www/bip  
 ERR\_4008 Przekroczony dopuszczalny rozmiar logo

## 4.2. getListToPerformElectronicProcedure – Pobranie list instytucji wykonującej elektronicznie procedurę

Metoda umożliwia pobranie listy instytucji wykonujących elektronicznie procedurę

### Nazwa funkcji

**getListToPerformElectronicProcedure**

### Parametry wejściowe

| Parametry wysyłane metodą GET |     |             |           |         |                         |
|-------------------------------|-----|-------------|-----------|---------|-------------------------|
| Nazwa zmiennej                | Typ | obowiązkowy | domyślnie | warunki | opis                    |
| procedure_alias               | int | tak         |           |         | Identyfikator procedury |

### Struktura danych wyjściowych

```
{
  "content" : {
    "rows_count" : < liczba wszystkich rekordów >,
    "institutions" : [ { < drzewo instytucji >
      "id" : < identyfikator instytucji >,
      "latitude" : < szerokość geograficzna >,
      "longitude" : < długość geograficzna >,
      "name" : < nazwa instytucji >,
      "street" : < ulica >,
      "number" : < numer >,
      "post_code" : < kod pocztowy >,
      "city" : < miasto, kraj jeśli inny niż Polska >
    } ]
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

### Przykład wywołania

[http://\[adres\\_usługi\]/ ApiInstitutions/getListToPerformElectronicProcedure?procedure\\_id=1048](http://[adres_usługi]/ ApiInstitutions/getListToPerformElectronicProcedure?procedure_id=1048)

### Opis kodów

OK\_0000 poprawna odpowiedź  
 OK\_0002 poprawna odpowiedź nie odnaleziono wyników spełniających podane kryterium  
 ERR\_0004 - Niepoprawny typ pola



ERR\_0011 - pole [nazwa\_pola] jest wymagane.

### 4.3. getCategories - Pobranie listy kategorii instytucji

Metoda umożliwiająca pobranie listy kategorii instytucji, które nie są typem specjalnym

#### Nazwa funkcji

**getCategories**

#### Parametry wejściowe

brak

#### Struktura danych wyjściowych

```
{
  "content" : {
    "UrządTyp" : < typ instytucji >,
    "id" : < id kategorii >,
    "name" : < nazwa kategorii >
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

#### Przykład wywołania

Pobranie pełnej listy kategorii instytucji

[http://\[adres\\_usługi\]/ApiInstitutions/getCategories](http://[adres_usługi]/ApiInstitutions/getCategories)

#### Opis kodów

OK\_0000 poprawna odpowiedź

### 4.4. getHints Pobieranie podpowiedzi nazw instytucji

Metoda umożliwiająca pobranie nazw instytucji rozpoczynających się od podanego w parametrze słowa

#### Nazwa funkcji

**getHints**

#### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |                |                                      |
|-------------------------------|--------|-------------|-----------|----------------|--------------------------------------|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki        | opis                                 |
| search                        | string | tak         |           | max 100 znaków | początkowy fragment nazwy instytucji |

|       |     |     |    |                                  |                           |
|-------|-----|-----|----|----------------------------------|---------------------------|
| limit | int | nie | 10 | min wartość 1, max wartość 99999 | liczba wyników na stronie |
|-------|-----|-----|----|----------------------------------|---------------------------|

### Struktura danych wyjściowych

```
{
  "content" : {
    "name" : < nazwa instytucji >
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

### Przykład wywołania

Pobranie podpowiedzi 5 instytucji, których nazwa rozpoczyna się od „Izba”

```
http://[adres_usługi]/ApiInstitutions/getHints?search=Izba&limit=5
```

Pobranie 10 (domyślna liczba) instytucji, których nazwa rozpoczyna się od „Izba”

```
http://[adres_usługi]/ApiInstitutions/getHints?search=Izba
```

### Opis kodów

OK\_0000 - poprawna odpowiedź

ERR\_0010 - Pole wykracza poza dopuszczalny zakres

ERR\_0004 - Niepoprawny typ pola

ERR\_0011 - Brak pola obowiązkowego

ERR\_0006 - Przekroczona dopuszczalna długość pola

## 4.5. getList - Pobieranie listy instytucji

Metoda umożliwiająca pobranie listy instytucji odpowiadających zadanym kryteriom. Nie zwraca instytucji o typie określonym jako typ specjalny. Jeżeli zdefiniowano parametr with\_departments jako AUTO, to jeżeli system w odpowiedzi na wprowadzony parametr search nie wyszuka żadnej pasującej instytucji, automatycznie przełączy się na wyszukiwanie po wydziałach instytucji tworząc wynikowo strukturę instytucja -> wydziały. Jeśli parametr with\_departments przyjmie wartość 1, pobrane zostaną instytucje posiadające wydziały w przypadku, kiedy nazwa instytucji lub wydziału pasuje do wyszukiwanej frazy. Jeśli dla instytucji odnalezionych zostanie więcej niż 1 wydział (dla with\_departments = 1), weight przyjmie wartość najlepiej dopasowanej pary instytucja-wydział na podstawie wyszukiwanej frazy. Gdy parametr zostanie ustawiony na 0, metoda zwróci zawsze tylko urzędy, bez struktury drzewiastej odpowiedzi.

### Nazwa funkcji

getList

### Parametry wejściowe

| Parametry wysyłane metodą GET |     |             |           |         |      |
|-------------------------------|-----|-------------|-----------|---------|------|
| Nazwa zmiennej                | Typ | obowiązkowy | domyślnie | warunki | opis |

|                   |        |     |   |  |  |
|-------------------|--------|-----|---|--|--|
| search            | string | nie |   | max 100 znaków   | wyszukiwana fraza  |
| group             | string | nie |   | na liście (CENTRALNE, SKARBOWE, SADY, ZUS, CECHY, UMG[=urzędy miast i gmin), POZOSTALE   | grupa kategorii instytucji                                   |
| category_id       | int    | nie |   |  | id kategorii instytucji                                      |
| voivodeship_teryt | string | nie |   | max 2 znaki  | teryt województwa w zakresie którego znajduje się instytucja |
| city              | string | nie |   | max 100 znaków   | miejsowość instytucji  |
| with_departments  | String | nie | 0   | na liście ( <p>0, &lt;brak wydziałów&gt;</p> <p>1, &lt;do listy dołączone będą wydziały pasujące do parametru wyszukiwania serach&gt;</p> <p>AUTO &lt;pobieranie listy bez wydziałów, a jeśli nie znaleziono żadnego, nastąpi wyszukanie wyników z uwzględnieniem nazw wydziałów</p> ) | czy dołączyć wydziały do wyników wyszukiwania                |
| page              | int    | nie | 1   | min wartość 1, max wartość 999   | numer strony wyszukiwania                                    |
| limit             | int    | nie | 10  | min wartość 1, max wartość 99999   | liczba wyników na stronie                                    |
| sort              | string | nie | NAME gdy brak parametru „search”, WEIGHT gdy wprowadzono "search" | na liście ( <p>'ID',</p> <p>'ELECTRONIC_PROCEDURES_COUNT',</p> <p>'EXECUTABLE_PROCEDURES_COUNT', 'CITY', 'NAME', 'NUMBER', 'POST_CODE', 'STREET', 'WEIGHT')</p> <p>   pole sortowania</p>  |  |
| order             | string | nie | asc   | na liście (asc, desc)  | kierunek sortowania  |
| range_teryt       | string | nie |   | max 7 znaków   | zasięg terytorialny  |

|     |     |     |  |   |   |
|-----|-----|-----|--|---|---|
| esp | int | nie |  | brak parametru – wszystkie instytucje<br>0 - instytucje bez ESP<br>1 – instytucje z ESP | filtrowanie instytucji po elektronicznej skrzynce podawczej |
|-----|-----|-----|--|---|---|

### Struktura danych wyjściowych

```
{
  "content" : {
    "rows_count" : < liczba rekordów ogółem (bez limitów) >,
    "institutions" : [ {
      "electronic_procedures_count" : < liczba procedur elektronicznych powiązanych z
instytucją >,
      "executable_procedures_count" : < liczba procedur wykonywanych powiązanych z instytucją
>,
      "id" : < id instytucji >,
      "name" : < nazwa instytucji >,
      "executive_name" : < nazwa organu wykonawczego >,
      "city" : < nazwa miejscowości, kraj jeśli inny niż Polska >,
      "number" : < numer lokalu >,
      "post_code" : < kod pocztowy >,
      "street" : < ulica >,
      "voivodeship" : < województwo >,
      "weight" : < waga sphinx (gdy podano parametr "search") >,
      ===== WERSJA V2 =====
      "country_code" : < kod kraju zgodny z ISO 3166 >
      ===== KONIEC WERSJI V2 =====

      "departments" : [ { < lista znalezionych wydziałów instytucji >
        "id" : < id wydziału >,
        "city" : < nazwa miejscowości, kraj jeśli inny niż Polska >,
        "name" : < nazwa wydziału >,
        "number" : < numer lokalu >,
        "post_code" : < kod pocztowy >,
        "street" : < ulica >,
        "voivodeship" : < województwo >,
        ===== WERSJA V2 =====
        "country_code" : < kod kraju zgodny z ISO 3166 >
        ===== KONIEC WERSJI V2 =====
      } ],
    } ],
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

### Przykłady wywołania

```
http://[adres_usługi]/ApiInstitutions/getList?search=izba&group=cechy&category_id=9&voivodeshi
p_teryt=28&city=Leszno&page=2&limit=5&sort=weight&order=asc
http://[adres_usługi]/ApiInstitutions/getList?search=Ministerstwo
```

### Opis kodów

OK\_0000 - poprawna odpowiedź

ERR\_0004 - Niepoprawny typ pola

ERR\_0006 - Przekroczona dopuszczalna długość pola

ERR\_0010 - Pole wykracza poza dopuszczalny zakres

ERR\_0011 - Brak pola obligatoryjnego

ERR\_0013 - Wartość nie znajduje się na liście  
 ERR\_0014 - Pole nie będące ID  
 ERR\_0035 - Żądana strona wykracza poza listę wyników

#### 4.6. getDetails - Pobieranie szczegółów instytucji

Metoda umożliwiająca pobranie szczegółów instytucji odpowiadających zadanym kryteriom.

#### Nazwa funkcji

getDetails

#### Parametry wejściowe

| Parametry wysyłane metodą GET |     |                                  |           |         |  |
|-------------------------------|-----|----------------------------------|-----------|---------|--|
| Nazwa zmiennej                | Typ | obowiązkowy                      | domyślnie | warunki | opis                                       |
| id                            | int | tak (w przypadku niepodania esp) |           |         | identyfikator instytucji                   |
| esp                           | int | tak (w przypadku niepodania id)  |           |         | elektroniczna skrzynka podawcza instytucji |

#### Struktura danych wyjściowych

```
{
  "content" : {
    "Institution" : {
      "electronic_procedures_count" : < liczba procedur elektronicznych powiązanych z
instytucją >,
      "executable_procedures_count" : < liczba procedur wykonywanych powiązanych z instytucją
>,
      "id" : < id instytucji >,
      "name" : < nazwa instytucji >,
      "executive_name" : < nazwa organu wykonawczego >
    },
    "InstitutionType" : {
      "type_id" : < identyfikator typu instytucji >,
      "type_name" : < nazwa typu instytucji >
    },
    "InstitutionDetails" : {
      "shortcut" : < skrót instytucji >,
      "nip" : < NIP >,
      "regon" : < REGON >,
      "street" : < ulica >,
      "number" : < numer lokalu >,
      "post_code" : < kod pocztowy >,
      "city" : < miejscowość, kraj jeśli inny niż Polska >,
      "voivodeship_teryt_id" : < identyfikator terytu >,
      "working_hours" : < godziny pracy instytucji >,
      "additional_info" : < dodatkowe informacje; m.in. zakres działalności >,
      "email" : < email >,
      "phone" : < telefon >,
      "fax" : < fax >,
      "www" : < strona www >,
      "bip" : < strona bip >,
      "logo" : < herb/logo instytucji >,
      "latitude" : < szerokość geograficzna >,
      "longitude" : < długość geograficzna >,
    }
  }
}
```

```

    "esp" : < nazwa skrzynki podawczej; tymczasowo nazwa epuap >,
    "epuap_id" : < identyfikator instytucji w epuap >,
    "general_eservice" : < adres e-usługi pisma ogólnego >,
    "responsible_procedures_count" : < liczba procedur za które instytucja jest
odpowiedzialna >,
    "queue_system_link" : < link do systemu kolejkowego >,
    "queue_system_description" : < opis systemu kolejkowego >
===== WERSJA V2 =====
    "country_code" : < kod kraju zgodny z ISO 3166 >
===== KONIEC WERSJI V2 =====

},
    "InstitutionDepartments" : [ { < wydziały instytucji >
        "Department" : {
            "id" : < identyfikator wydziału >,
            "name" : < nazwa wydziału >
        }
    } ],
    "InstitutionPkd" : [ {
        "pkd_kod" : < kod pkd >,
        "pkd_opis" : < opis pkd >
    } ],
    "InstitutionTerritoryRange" : [ {
        "TerritoryRange" : {
            "voivodeship" : < województwo >,
            "district" : < powiat >,
            "community" : < gmina >,
            "community_kind" : < rodzaj gminy >,
            "teryt" : < kod teryt >
        }
    } ]
},
    "language" : < kod języka >,
    "status" : {
        "code" : < status odpowiedzi >,
        "message" : < komunikat >
    }
}

```

### Przykłady wywołania

```

http://[adres_uslugi]/ApiInstitutions/getDetails?id=1322
http://[adres_uslugi]/ApiInstitutions/getDetails?esp=/3825aby1g5/SkrytkaESP

```

### Opis kodów

```

OK_0000 - poprawna odpowiedź

ERR_0011 - Brak pola obligatoryjnego
ERR_0014 - Pole nie będące ID
ERR_0033 - Instytucja o podanym ID nie istnieje
ERR_0051 - Jedno z 2 pól: id, esp jest wymagane
ERR_0052 - Instytucja o podany ESP nie istnieje

```

## 4.7. getDepartmentDetails - Pobieranie szczegółów wydziału instytucji

Metoda umożliwiająca pobranie szczegółów wydziału danej instytucji

### Nazwa funkcji

**getDepartmentDetails**

## Parametry wejściowe

| Parametry wysyłane metodą GET |     |             |           |         |                                   |
|-------------------------------|-----|-------------|-----------|---------|-----------------------------------|
| Nazwa zmiennej                | Typ | obowiązkowy | domyślnie | warunki | opis                              |
| id                            | int | tak         |           |         | identyfikator wydziału instytucji |

## Struktura danych wyjściowych

```
{
  "content" : {
    "Department" : {
      "electronic_procedures_count" : < liczba procedur elektronicznych powiązanych z
      wydziałem >,
      "executable_procedures_count" : < liczba procedur wykonywanych powiązanych z wydziałem
      >,
      "id" : < id wydziału instytucji >,
      "name" : < nazwa wydziału instytucji >
    },
    "DepartmentDetails" : {
      "shortcut" : < skrót >,
      "street" : < ulica >,
      "number" : < numer lokalu >,
      "post_code" : < kod pocztowy >,
      "city" : < miejscowość, kraj jeśli inny niż Polska >,
      "voivodeship_teryt_id" : < identyfikator terytu >,
      "working_hours" : < godziny pracy wydziału >,
      "additional_info" : < dodatkowe informacje, m.in. zakres działalności >,
      "email" : < email >,
      "phone" : < telefon >,
      "fax" : < fax >,
      "www" : < strona www >,
      "bip" : < strona bip >,
      "latitude" : < szerokość geograficzna >,
      "longitude" : < długość geograficzna >,
      "is_queue_system_central" : < czy wydział korzysta z systemu kolejkowego instytucji, do
      której należy >,
      "queue_system_link" : < link do systemu kolejkowego >,
      "queue_system_description" : < opis systemu kolejkowego >,
      ===== WERSJA V2 =====
      "country_code" : < kod kraju zgodny z ISO 3166 >
      ===== KONIEC WERSJI V2 =====
    },
    "ParentInstitution" : {
      "id" : < id instytucji >,
      "name" : < nazwa instytucji >,
      "general_eservice" : < adres e-usługi pisma ogólnego instytucji >,
      "logo" : < herb/logo instytucji >,
    },
    "InstitutionTerritoryRange" : [ {
      "TerritoryRange" : {
        "voivodeship" : < województwo >,
        "district" : < powiat >,
        "community" : < gmina >,
        "community_kind" : < rodzaj gminy >,
        "teryt" : < kod teryt >
      }
    } ]
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

```
}  
}
```

### Przykłady wywołania

```
http://[adres_usługi]/ApiInstitutions/getDepartmentDetails?id=1322
```

### Opis kodów

OK\_0000 - poprawna odpowiedź

ERR\_0011 - Brak pola obligatoryjnego

ERR\_0014 - Pole nie będące ID

ERR\_0032 - Wydział o podanym ID nie istnieje

## 4.8. getListByTeryt - Pobieranie listy jednostek samorządu terytorialnego

Metoda umożliwiająca pobranie listy jednostek samorządu terytorialnego (starostwa powiatowe, urzędy marszałkowskie, urzędy wojewódzkie, urzędy miast i gmin) dla wybranego obszaru teryt

Nazwa funkcji

### getListByTeryt

Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |              |           |
|-------------------------------|--------|-------------|-----------|--------------|-----------|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki      | opis      |
| teryt                         | string | tak         |           | max 6 znaków | kod teryt |

Struktura danych wyjściowych

```
{  
  "content" : {  
    "electronic_procedures_count" : < liczba procedur elektronicznych powiązanych z  
instytucją >,  
    "executable_procedures_count" : < liczba procedur wykonywanych powiązanych z instytucją  
>,  
    "id" : < id instytucji >,  
    "city" : < nazwa miejscowości >,  
    "name" : < nazwa instytucji >,  
    "number" : < numer lokalu >,  
    "post_code" : < kod pocztowy >,  
    "street" : < ulica >,  
  },  
  "language" : < kod języka >,  
  "status" : {  
    "code" : < status odpowiedzi >,  
    "message" : < komunikat >  
  }  
}
```

### Przykłady wywołania

```
http://[adres_usługi]/ApiInstitutions/getListByTeryt?teryt=306402
```

### Opis kodów

OK\_0000 - poprawna odpowiedź

ERR\_0006 - Przekroczona dopuszczalna długość pola



ERR\_0011 - Brak pola obligatoryjnego

## 5. ApiProcedures

### 5.1. Wprowadzenie

#### 5.1.1. Zastosowanie parametru ALIAS zamiast ID w wywołaniach

Główne wywołanie procedur opiera się o tzw. **ALIAS** czyli parametr składający się standardowo z głównego, statycznego identyfikatora procedury, dawniej identyfikatora UEPA, niezależnego od wersji, oraz nazwy skróconej procedury pozbawionej znaków specjalnych, spacji i znaków diakrytycznych. Nazwa może zmieniać się w kolejnych wersjach publikowanej procedury. Jeżeli alias jest parametrem wejściowym dla metod, system dopuszcza podanie jedynie ID nie badając pozostałego ciągu znaków. Wyjątkiem od tej reguły jest wywołanie niestandardowe dla `getDetails`, gdzie podobnie jak dla innych części systemu (sytuacje życiowe, publikacje) można wymusić wyświetlenie konkretnej wersji procedury niezależnie od jej statusu. Alias wówczas zawiera prefix „TEST”, ID wersji procedury, znacznik czasu, prawidłowy alias.

Portal w celu zachowania poprawnych linków, oraz co za tym idzie poprawności indeksowania w wyszukiwarkach powinien w przypadku wywołania procedury z niekompletnym lub niewłaściwym aliasem automatycznie przesyłać do przeglądarki żądanie zmiany 301 (Move Permanently) na właściwy alias. Bieżący, właściwy alias zawsze znajduje się w odpowiedziach API.

#### Przykład wywołania API `getDetails` z różnym zastosowaniem parametru ALIAS

##### Wywołanie standardowe

```
http://[adres_usługi]/ ApiProcedures/getDetails?alias=111-wpis-do-rejestru-operatorow-pocztowych
```

##### Wywołanie bez nazwy aliasu

```
http://[adres_usługi]/ ApiProcedures/getDetails?alias=111
```

##### Wywołanie elementu o konkretnym id wersji

```
http://[adres_usługi]/ApiProcedures/getDetails?alias=*TEST-339-1413981229*111-wpis-do-rejestru-operatorow-pocztowych
```

### 5.2. `getHints` - Pobieranie odpowiedzi nazw procedur

Metoda umożliwiająca pobranie nazw procedur zawierających podany w parametrze ciąg znaków

#### Nazwa funkcji

`getHints`

#### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |                                  |                                      |
|-------------------------------|--------|-------------|-----------|----------------------------------|--------------------------------------|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki                          | opis                                 |
| search                        | string | tak         |           | max 100 znaków                   | fragment nazwy procedury             |
| limit                         | int    | nie         | 10        | min wartość 1, max wartość 99999 | maksymalna liczba zwróconych wyników |

|                 |        |     |    |                  |   |
|-----------------|--------|-----|----|------------------|---|
| lang            | string | nie | PL | kod języka       | język, w jakim mają być zwrócone procedury  |
| electronic_only | int    | nie | 0  | na liście (0, 1) | czy mają być zwrócone tylko procedury elektroniczne                                   |
| institution_id  | int    | nie |    | max len 20       | zwracane nazwy procedury są w kontekście instytucji realizującej o podanym id         |
| department_id   | int    | nie |    | max len 20       | zwracane nazwy procedur są w kontekście wydziału instytucji realizującej o podanym id |

### Struktura danych wyjściowych

```
{
  "content" : {
    "name" : < nazwa procedury >
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

### Przykład wywołania

Pobranie podpowiedzi 5 procedur elektronicznych, zawierających słowo "zezwolenie"

```
http://[adres_uslugi]/ApiProcedures/getHints?search=zezwolenie&limit=5
```

Pobranie 10 (domyślna liczba) procedur w języku angielskim, zawierającej frazę "trans"

```
http://[adres_uslugi]/ApiInstitutions/getHints?search=transport&lang=EN
```

Pobranie 10 (domyślna liczba) procedur w języku angielskim, zawierającej frazę "trans" dla instytucji o id = 110

```
http://[adres_uslugi]/ApiInstitutions/getHints?search=transport&lang=EN&institution_id=110
```

Pobranie 10 (domyślna liczba) procedur w języku angielskim, zawierającej frazę "trans" dla wydziału o id = 923

```
http://[adres_uslugi]/ApiInstitutions/getHints?search=transport&lang=EN&department_id=923
```

### Opis kodów

OK\_0000 - poprawna odpowiedź  
OK\_0002 - poprawna odpowiedź, brak wyników

ERR\_0004 - Niepoprawny typ pola  
ERR\_0005 - Niepoprawny kod języka  
ERR\_0006 - Przekroczona dopuszczalna długość pola  
ERR\_0010 - Pole wykracza poza dopuszczalny zakres  
ERR\_0011 - Brak pola obowiązkowego  
ERR\_0013 - Element nie znajduje się na liście

## 5.3. getList - Pobieranie listy procedur

Metoda umożliwiająca pobranie listy procedur wg zadanych kryteriów wyszukiwania. W przypadku podania parametru lang innego niż 'PL' oraz braku parametrów 'search' i 'letter' pobierana jest lista

procedur w języku obcym odpowiadająca kryteriom, a w przypadku braku opublikowanej wersji tłumaczenia zwracana jest wersja polska procedury. W sytuacji, gdy podany jest parametr `electronic_only` i parametr `institution_id`, otrzymamy w wyniku procedury, które są elektroniczne w kontekście podanego urzędu. Wartość pola `eservice` nie uwzględnia kontekstu instytucji realizującej.

## Nazwa funkcji

`getList`

## Wersja

V4

## Parametry wejściowe

| Parametry wysyłane metodą GET       |        |             |           |  |   |
|-------------------------------------|--------|-------------|-----------|--|---|
| Nazwa zmiennej                      | Typ    | obowiązkowy | domyślnie | warunki  | opis  |
| <code>search</code>                 | string | nie         |           | max 300 znaków   | nazwa procedury   |
| <code>pkd</code>                    | string | nie         |           |  | lista kodów pkd powiązanych z procedurami, oddzielona przecinkami   |
| <code>type</code>                   | string | nie         |           | na liście (TYP_UZ_KW_ZA, TYP_SW_US_TR, TYP_ZA_REG, TYP_DZ_REG, TYP_ZE_PO_FO_REGL, TYP_INN) | lista typów procedury oddzielona przecinkami.<br>Tłumaczenie kodów:<br>TYP_UZ_KW_ZA - Uznawanie kwalifikacji zawodowych;<br>TYP_SW_US_TR - Świadczenie usług transgranicznych;<br>TYP_ZA_REG - Zawody, w tym regulowane;<br>TYP_DZ_REG - Działalność regulowana;<br>TYP_ZE_PO_FO_REGL - Zezwolenia i pozostałe formy reglamentacji;<br>TYP_INN - Inne |
| <code>starting_business_only</code> | int    | nie         | 0         | na liście (0, 1)   | czy wyszukiwać tylko procedury związane z zakładaniem działalności gospodarczej (działa dla v2)   |

|                    |        |     |      |  |  |
|--------------------|--------|-----|------|--|--|
| electronic_only    | int    | nie | 0    | na liście (0, 1)   | czy wyszukiwać tylko procedury elektroniczne   |
| category_id        | int    | nie |      |  | id kategorii procedury   |
| profession_id      | int    | nie |      |  | id zawodu  |
| letter             | string | nie |      |  | litera, od której rozpoczyna się nazwa procedury   |
| lang               | string | nie | PL   | kod języka   | język tłumaczenia procedury  |
| institution_id     | int    | nie |      |  | id instytucji wykonującej powiązanej z procedurami   |
| page               | int    | nie | 1    | min wartość 1, max wartość 999   | numer strony wyszukiwania  |
| limit              | int    | nie | 10   | min wartość 1, max wartość 99999   | liczba wyników na stronie  |
| sort               | string | nie | NAME | na liście (ID, ELECTRONIC, NAME, WEIGHT, PUBLICATION_DATE, MODIFICATION_DATE)                        | pole sortowania  |
| order              | string | nie | ASC  | na liście (ASC, DESC)  | kierunek sortowania  |
| department_id      | int    | nie |      |  | Id wydziału powiązanego z procedurami  |
| epk_only           | int    | nie | 0    | na liście (0, 1)   | czy wyszukiwać tylko procedury, które posiadają e-usługę typu EPK (zawierającą w adresie url fragment biznes.gov.pl) |
| central_only       | int    | nie | 0    | na liście (0, 1)   | czy wyszukiwać tylko procedury, które mają e-usługę nadaną centralnie  |
| business_lifecycle | string | nie |      | na liście (starting, doing, closing); można wybrać więcej niż jedną opcję oddzielając je przecinkiem | zawężanie wyników wyszukiwania wg kategorii – zakładania, prowadzenia i zamykania działalności                       |

### Struktura danych wyjściowych

```
{
  "content" : {
    "procedures" : [ {
      "electronic" : < czy procedura elektroniczna >,
      "id" : < id procedury >,

```

```

"alias"< alias procedury >,
"name" : < nazwa procedury >,
"weight" : < waga >,
"language" : < kod języka procedury >,
"modification_date" : < data modyfikacji >,
"short_introduction" : < zawartość pola „Krótkie wprowadzenie” >,
===== WERSJA V4 =====

"eservice" : [ { < usługi elektroniczne - Wartość pola nie uwzględnia kontekstu
instytucji realizującej>
  "link" : < link do usługi >,
  "description" : < opis usługi >,
  "category" : < kod określający typ usługi >
} ],
===== KONIEC WERSJI V4 =====
"rows_count" : < liczba rekordów ogółem >
},
"language" : < kod języka >,
"status" : {
  "code" : < status odpowiedzi >,
  "message" : < komunikat >
}
}

```

### Przykład wywołania

Pobranie listy procedur bez kryteriów, z domyślną liczbą wyników 10

```
http://[adres_usługi]/ApiProcedures/getList
```

Pobranie listy 50 procedur elektronicznych

```
http://[adres_usługi]/ApiProcedures/getList?limit=50&electronic_only=1
```

Pobranie listy procedur powiązanych z zawodem o id 8 rozpoczynających się na literę A

```
http://[adres_usługi]/ApiProcedures/getList?profession_id=8&letter=a
```

### Opis kodów

```

OK_0000 - poprawna odpowiedź
OK_0002 - poprawna odpowiedź, brak wyników

ERR_0004 - Niepoprawny typ pola
ERR_0005 - Niepoprawny kod języka
ERR_0006 - Przekroczona dopuszczalna długość pola
ERR_0010 - Pole wykracza poza dopuszczalny zakres
ERR_0013 - Element nie znajduje się na liście
ERR_0014 - Pole nie jest ID
ERR_0023 - Niepoprawna lista PKD
ERR_0024 - Niepoprawna lista typów procedury
ERR_0035 - Żądana strona wykracza poza listę wyników

```

## 5.4. getElectronicList - Pobieranie listy procedur elektronicznych

Metoda umożliwiająca pobranie listy procedur elektronicznych wg zadanych kryteriów wyszukiwania wraz z powiązаныmi instytucjami

### Nazwa funkcji

**getElectronicList**

## Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |   |   |
|-------------------------------|--------|-------------|-----------|---|---|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki   | opis  |
| search                        | string | nie         |           | max 100 znaków  | nazwa procedury   |
| pkd                           | string | nie         |           |   | lista kodów pkd powiązanych z procedurami, oddzielona przecinkami   |
| type                          | string | nie         |           | na liście (DEF_CSU, DEF_DR, DEF_SUTG, DEF_UKZ, DEF_ZIIFR, DEF_ZR) | lista typów procedury oddzielona przecinkami. Tłumaczenie kodów: DEF_CSU - Czasowe Świadczenie Usług; DEF_DR - Działalność Regulowana; DEF_SUTG - Świadczenie Usług Transgranicznych; DEF_UKZ - Uznawanie Kwalifikacji Zawodowych; DEF_ZIIFR - Zezwolenia I Inne Formy Reglamentacji; DEF_ZR - Zawody, w tym Regulowane |
| lang                          | string | nie         | PL        | kod języka  | język tłumaczenia procedury   |
| page                          | int    | nie         | 1         | min wartość 1, max wartość 999                                    | numer strony wyszukiwania   |
| limit                         | int    | nie         | 10        | min wartość 1, max wartość 99999                                  | liczba wyników na stronie   |
| sort                          | string | nie         | NAME      | na liście (ID, ELECTRONIC, NAME, WEIGHT)                          | pole sortowania   |
| order                         | string | nie         | ASC       | na liście (ASC, DESC)   | kierunek sortowania   |

## Struktura danych wyjściowych

```
{
  "content" : {
    "procedures" : [ {
      "electronic" : < czy procedura elektroniczna >,
      "id" : < id procedury >,
      "alias" : < alias procedury >,
      "name" : < nazwa procedury >,
      "weight" : < waga >,
      "language" : < kod języka >,
      "Institutions" : [ {
        "inst_id" : < id instytucji >,
        "inst_nazwa" : < nazwa instytucji >
      } ]
    } ]
  },
  "rows_count" : < liczba rekordów ogółem >
},
"language" : < kod języka >,
"status" : {
  "code" : < status odpowiedzi >,
  "message" : < komunikat >
}
}
```

## Przykład wywołania

Pobranie listy procedur elektronicznych bez kryteriów, z domyślną liczbą wyników 10

```
http://[adres_usługi]/ApiProcedures/getElectronicList
```

Pobranie listy 50 procedur elektronicznych zawierających frazę "zewz"

```
http://[adres_usługi]/ApiProcedures/getElectronicList?limit=50&search=zewz
```

## Opis kodów

OK\_0000 - poprawna odpowiedź  
OK\_0002 - poprawna odpowiedź, brak wyników  
  
ERR\_0004 - Niepoprawny typ pola  
ERR\_0005 - Niepoprawny kod języka  
ERR\_0006 - Przekroczona dopuszczalna długość pola  
ERR\_0010 - Pole wykracza poza dopuszczalny zakres  
ERR\_0013 - Element nie znajduje się na liście  
ERR\_0023 - Niepoprawna lista PKD  
ERR\_0024 - Niepoprawna lista typów procedury  
ERR\_0035 - Żądana strona wykracza poza listę wyników

## 5.5. getDetails - Pobieranie szczegółów procedury

Metoda pobierająca szczegóły procedury. W wyniku otrzymamy szczegóły procedury wraz z informacją czy procedurę można wykonać elektronicznie oraz ewentualnie urzędem wykonującym procedurę.

### Nazwa funkcji

getDetails

### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |                  |   |
|-------------------------------|--------|-------------|-----------|------------------|---|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki          | opis  |
| alias                         | string | tak         |           | max 255 znaków   | alias procedury (id-name_abbreviation)  |
| institution_id                | int    | nie         |           | ?                | identyfikator instytucji  |
| teryt                         | string | nie         |           | ?                | kod TERYT miejscowości, dla której ma wyświetlany być opis procedury  |
| lang                          | string | nie         | PL        | ?                | język opisu procedury   |
| type                          |        | nie         |           | ?                | dla kogo ta procedura   |
| check_if_anyone_has_epuap     | int    | nie         | 0         | na liście (0, 1) | Metoda przy włączonym parametrze sprawdza, czy jakkolwiek instytucja w obrębie instytucji wykonujących dla tej procedury posiada wprowadzony identyfikator podmiotu ePUAP. Informacja wyświetlana jest w wyniku w polu "anyone_has_epuap" |



|     |        |     |  |        |  |
|-----|--------|-----|--|--------|--|
| sym | string | nie |  | 7 cyfr | Kod SYM z bazy SIMC miejscowości, dla której ma być wyświetlony opis procedury |
|-----|--------|-----|--|--------|--|

### Struktura danych wyjściowych

```
{
  "content" : {
    "procedure_name" : {
      "name" : < nazwa procedury >,
      "name_abbreviation" : < skrót nazwy procedury >,
      "keywords" : < słowa kluczowe >,
      "keywords_additional" : < słowa kluczowe dodatkowe >,
      "alias" : < poprawny alias procedury >,
      "id" : < id procedury >,
      "version_id" : < id wersji procedury >,
      "first_publication_date" : < data pierwszej publikacji >,
      "last_modification_date" : < data ostatniej modyfikacji >,
      "is_expired" : < czy procedura jest oznaczona jako nieaktualna (true/false) >,
      =====WERSJA V5=====
      "expiration_date" : < data oflagowania opisu usługi jako nieaktualny >,
      =====KONIEC WERSJI V5=====
      "confirmation_date" : < data zatwierdzenia opisu procedury przez Instytucję Centralną
    >,
      "verification_date" : < data świadomego potwierdzenia aktualności całego opisu
procedury przez Instytucję Centralną >,
      "recall_date" : < data planowanego wycofania >,
      "responsible_institution_id" : < identyfikator instytucji odpowiedzialnej za opis
procedury >,
      "responsible_institution_name" : < nazwa instytucji >,
      "publication_status_code" : < kod procedury (ETA; EZA; IZA) >,
      "publication_status_code_description" : < opis do pola code >,
      "accepting_institution_id" : < identyfikator instytucji odpowiedzialnej za opis
procedury >,
      "accepting_institution_name" : < nazwa instytucji >,
      "institution_verification_date" : < data przekazania do weryfikacji instytucji >
    },
    "procedure_execution_type" : < rodzaj procedury ze względu na instytucję wykonującą
(CENTRAL-procedura wykonywana przez jedną instytucję centralną; RANGE-procedura wykonywana
przez instytucję przypisaną do wybranego obszaru >,
    "procedure_execution" : {
      "institution" : { < dane wskazanej instytucji w której można zrealizować procedurę >
        "id" : < id instytucji >,
        "name" : < nazwa instytucji >,
        "street" : < ulica >,
        "number" : < numer domu >,
        "post_code" : < kod pocztowy >,
        "city" : < miasto, kraj jeśli inny niż Polska >,
        "voivodeship_teryt" : < teryt województwa >,
        "voivodeship" : < nazwa województwa >,
        =====WERSJA V6=====
        "email" : < adres email >,
        "phone" : < numer telefonu >,
        =====KONIEC WERSJI V6=====
      }
      "department" : { < wydział instytucji >
        "id" : < id wydziału >,
        "name" : < nazwa wydziału >,
        "street" : < ulica >,
        "number" : < numer domu >,
        "post_code" : < kod pocztowy >,
        "city" : < miasto, kraj jeśli inny niż Polska >,
        "voivodeship_teryt" : < teryt województwa >
        =====WERSJA V6=====
        "email" : < adres email >,

```

```

"phone" : <numer telefonu >,
=====KONIEC WERSJA V6=====
}
},
===== WERSJA V2 =====
"eservices" : [ { < usługi elektroniczne >
  "link" : < link do usługi >,
  "description" : < opis usługi >,
  "category" : < kod określający typ usługi (epk-usługa specjalna;
executing_institution-usługa zdefiniowana przez instytucję realizującą; central-usługa
centralna; general_eservice - usługa oparta o pismo ogólne >
} ]
===== WERSJA V3 =====

"how_to_execute" : { < w jaki sposób można zrealizować usługę >
  "options" : [ <lista opcji sposobów realizacji usługi - podzbiór INSTITUTION, MAIL,
ONLINE> ],
  "additional_description" : < dodatkowy opis który jeśli niepusty to można wykorzystać
do opisanie innych sposobów załatwienia sprawy lub barier w powyższych sposobach >
  "eservices" : [ { < usługi elektroniczne >
    "link" : < link do usługi >,
    "description" : < opis usługi >,
    "category" : < kod określający typ usługi (epk-usługa specjalna;
executing_institution-usługa zdefiniowana przez instytucję realizującą; central-usługa
centralna; general_eservice - usługa oparta o pismo ogólne >
} ]
},
"room_person" : < pokój / osoba >
===== KONIEC WERSJA V3 =====

"anyone_has_epuap" : < czy istnieje min. jedna instytucja z podmiotem ePUAP (TRUE, FALSE,
NULL - jeśli metoda wywołana bez parametru 'check_if_anyone_has_epuap') >
},
"procedure_general" : { < dane ogólne o procedurze >
  "description_code" : < typ opisu procedury (OPIS_PELNY, OPIS_SKROCONY) >
  "type" : < typ procedury >,
  "purpose" : < cel procedury >,
  "statutory_duration" : { < ustawowy czas realizacji procedury >
    "code" : < kod typu czasu realizacji >,
    "code_description" : < opis kodu >,
    "description" : < opis gdy kod typu ustawiony na opis >
  },
  "has_fees" : < czy procedura posiada opłaty (true/false), wartość pola uzależniona jest
od wartości pola description_code; gdy description_code == OPIS_SKROCONY to has_fees = null;
gdy description_code == OPIS_PELNY to has_fees = (true/false) >,
  "pkd" : {
    "status_code" : < kod statusu PKD (NIE_DOTYCZY; WYBRANE; WSZYSTKIE >,
    "status_description" : < opis statusu pobrany ze słownika (np. Procedura dotyczy
wybranych gałęzi PKD) >,
    "list" : [ { < tablica obiektów PKD >
      "code" : < kod PKD >,
      "title" : < nazwa gałęzi PKD >,
      "hint" : < odpowiedź do PKD >,
      "includes_description" : < opis informujący czego dotyczy PKD >,
      "not_includes_description" : < opis informujący czego nie dotyczy PKD >
    } ],
  },
  "participants" : { < tablica uczestników >
    "name" : < wnioskodawca; nazwa >,
    "name_additional" : < nazwa dodatkowa >
  },
  "local_jurisdiction" : { < właściwość miejscowa >

```

```

        "for_entrepreneur" : < kod właściwości miejscowej
(ZASP_NIE_DOTYCZY; ZASP_MIE_WYK; ZASP_SIEDZ; ZASP_DOW_URZ; ZASP_LOK_PRZEDS;
ZASP_MIE_WYK_SIEDZ; ZASP_SIE_LUB_MIE; ZASP_LOK_SPRZED; ZASP_PORT_ZALAD; ZASP_MIE_POL_OB;
ZASP_MIE_WYK_RO; ZASP_MIE_POL_UPR; ZASP_POL_PLA_NAS; ZASP_PORT_MAC; ZASP_MIE_PROW_POL;
ZASP_KORZ_SRO; ZASP_ODP_KOM; ZASP_MIE_ZAM; ZASP_MIE_INW_BUD; ZASP_MIE_ROZ; ZASP_POL_BUD;
ZASP_LOK_ZJA; ZASP_LOK_ZAJ_PDROG; ZASP_POL_NIE; ZASP_POL_DRZE; ZASP_LOK_INST;
ZASP_MIE_POL_ZAB; ZASP_MIE_WYZBPR_ODP; ZASP_MIE_IMP_MAS; ZASP_URZ_SKR_PRZE; ZASP_MIE_ODP_CEL;
ZASP_MIE_PRO_KSI_RAC; ZASP_TRA_PRZEJ; ZASP_SIEDZ_URZ) >,
        "for_citizen" : < kod właściwości miejscowej
(ZASO_NIE_DOTYCZY; ZASO_MIE_ZAMEL; ZASO_MIE_ZAMIE; ZASO_DOW_URZ; ZASO_LOK_PRZEDS;
ZASO_MIE_ZAMEL_ZAMIE; ZASO_MIE_WYK_ZAW; ZASO_MIE_POB_CUDZ; ZASO_MIE_WYK_ROB; ZASO_MIE_POL_UPR;
ZASO_POL_PLA_NAS; ZASO_PORT_MAC; ZASO_MIE_PRO_POL; ZASO_KORZ_SRO; ZASO_ODP_KOM;
ZASO_MIE_INW_BUD; ZASO_MIE_ROZ; ZASO_POL_BUD; ZASO_LOK_ZJA; ZASO_LOK_ZAJ_PDROG; ZASO_POL_NIE;
ZASO_POL_DRZE; ZASO_LOK_INST; ZASO_MIE_POL_ZAB; ZASO_MIE_IMP_MAS; ZASO_MIE_ODP_CEL;
ZASO_MIE_POL_OBI; ZASO_TRA_PRZEJ; ZASO_SIEDZ_URZ) >
    },
    },
    "institutions_implementing_hint" : < odpowiedź dla użytkownika dotycząca wyboru
instytucji >,
    "institutions_implementing_in_range" : < { < lista id instytucji wykonujących procedurę
dla wskazanego obszaru (wymaga podania kodu teryt w parametrach wejściowych) lub null gdy
przypisana jest instytucja o typie specjalnym >
        "id" : < id instytucji >,
        "name" : < nazwa instytucji >
    } ],
    "institutions_implementing_all" : { < lista id instytucji i typów instytucji
realizujących procedurę wskazanych na formularzu opisu procedury >
        "units" : [ {
            "id" : < id instytucji >,
            "name" : < nazwa instytucji >
        } ],
        "types" : [ { < null gdy została przypisana instytucja o typie specjalnym >
            "id" : < id typu instytucji >,
            "name" : < nazwa typu instytucji (np. urzędy miast i gmin) >
        } ],
    },
    "procedure_steps" : { < tablica kroki procedury >
        "id" : < identyfikator >,
        "description" : < opis >,
        "number" : < numer kroku >,
        "type_code" : < kod typu kroku (W - złożenie wniosku; B - inny) >,
        "type_description" : < typ kroku (złożenie wniosku; inny) >,
        "title" : , < tytuł >
        "details" : {
            "documents" : {
                "documents_description" : < opis dotyczący dokumentów podpiętych pod procedurę >,
                "list" : [ { < lista dokumentów >
                    "title" : < tytuł dokumentu >,
                    "description" : < opis dokumentu >,
                    "id" : < identyfikator dokumentu >,
                    "context_code" : < kontekst dokumentu: WE-wejściowy, WY-wyjściowy, ZA-załącznik
>,
                "related_procedures" : [ { < lista procedur powiązanych poprzez dokument >
                    "id" : < identyfikator procedury >,
                    "alias" : < alias powiązanej procedury >,
                    "name" : < nazwa procedury >
                } ],
                "attachments" : {
                    "central" : [ { < lista załączników centralnych >
                        "name" : < nazwa załącznika >,
                        "type_code" : < typ załącznika (WZOR_PAPIEROWY; WZOR_ELEKTRONICZNY;
PRZYKLAD_ELEKTRONICZNY; PRZYKLAD_PAPIEROWY; INSTRUKCJA) >,
                        "type_description" : < opis kodu załącznika (Wzór papierowy; Wzór
elektroniczny; Przykład elektroniczny; Przykład papierowy; Instrukcja) >,

```

```

        "link" : < link do załącznika >,
        "visible" : < true/false >,
=====WERSJA V7=====
        "file_type" : < typ pliku >,
=====KONIEC V7=====
    } ],
    "executing_institution" : [ {
        "name" : < nazwa załącznika >,
        "type_code" : < typ załącznika, opis typów j.w. >,
        "type_description" : < typ opisu >,
        "link" : < link do załącznika >,
        "visible" : < true/false >,
=====WERSJA V7=====
        "file_type" : < typ pliku >,
=====KONIEC V7=====
    } ],
    },
    "forms" : [ {
        "form_code" : < kod formy (Ko; Ut; Or; Pe; Dw; Uk) >,
        "form_description" : < opis formy >
    } ],
    } ],
    },
    "fees" : {
        "value" : < wartość >,
        "currency" : < waluta (PLN, EUR) >,
        "description" : < opis >,
        "type_code" : < kod określający status uzupełnienia opłaty (central-opłata ustalona
przez instytucję centralną; waiting_for_exec_inst-opłata ustalona przez instytucję realizującą
ale jeszcze nie uzupełniona przez tę instytucję; executing_institution-opłata ustalona przez
instytucję realizującą już wypełniona przez instytucję >,
        "bank_account_number" : < numer rachunku bankowego instytucji realizującej >,
    },
    "deadline" : {
        "code" : < opis; nie_dotyczy >,
        "code_description" : < opis kodu typu (opis; nie_dotyczy) >,
        "description" : < działa dla typu opis >
    },
    "participants" : [ < pole nie występuje w przypadku braku uczestników > {
        "name" : < nazwa >,
        "name_additional" : < nazwa typu; nazwa urzędu; null >
    } ],
    "eservices" : [ {
        "link" : < link do usługi >,
        "description" : < opis usługi >,
        "type_code" : < kod określający typ usługi (epk-usługa specjalna;
executing_institution-usługa zdefiniowana przez instytucję realizującą; central-usługa
centralna ) >
    } ]
    }
    },
    "acts" : [ { < tablica aktów >
        "id" : ,
        "name" : ,
        "link" : ,
        "type_code" : < kod aktu prawnego Us; Ro; Uch; Pu; In >,
        "type_description" : < opis kodu aktu prawnego (Ustawa; Rozporządzenie; Uchwała;
Przepisyunijne; Inny >,
        "references" : [ { < lista nawiązań do innych aktów prawnych >
            "id" : < id aktu prawnego >,
            "name" : < link > : ,
            "type_code" : < kod aktu prawnego >,
            "type_description" : < opis kodu aktu prawnego >
        } ]
    } ]
} ]

```

```

} ],
"conditions" : {
  "input_conditions" : < warunki wejściowe >,
  "result" : < rezultat procedury !!! Uwaga to pole jest nieużywane >
},
"additional_information" : { < dodatkowe informacje >
  "alternative_process" : < przebieg alternatywny !!! Uwaga to pole jest nieużywane >,
  "additional_information" : < dodatkowe informacje >,
  "prevention_means" : < środki odwoławcze >,
},
"references" : { < Odwołania do innych procedur; publikacji; poradników >
  "with_procedures" : [ { < lista procedur zbliżonych tematycznie >
    "id" : < id powiązanej procedury >,
    "alias" : < alias powiązanej procedury >,
    "name" : < nazwa powiązanej procedury >
  } ],
  "with_publications" : [ { < lista publikacji w których mowa o tej procedurze >
    "id" : < id powiązanej publikacji >,
    "alias" : < alias powiązanej publikacji >,
    "name" : < nazwa powiązanej publikacji >,
  } ],
  "with_lifesituations" : [ { < lista poradników w których mowa o tej procedurze >
    "id" : < id powiązanego poradnika >,
    "alias" : < alias powiązanego poradnika >,
    "name" : < nazwa powiązanego poradnika >
  } ],
  "business_lifecycle_name" : < nazwa działalności dla cyklu życia, np.: "Cykl życia
działalności związanej [np.: z wpisem do rejestru działalności szkoleniowej]" >,
  "business_lifecycle" : [ { < lista procedur związana z cyklem życia działalności >
    "type" : < kod podtypu na procedurze: rozpoczęcie, prowadzenie, zakończenie
działalności (ROZ; PRO; ZAK) >,
    "id" : < id powiązanej procedury >,
    "alias" : < alias powiązanej procedury >,
    "name" : < nazwa powiązanej procedury >
  } ]
},
===== WERSJA V4 =====
"localization_name" : <mięscowość dla której wyświetlana jest procedura, nazwa
miejsowości wyświetlana użytkownikowi w standardzie: miejscowość (gmina <nazwa>, woj.
<NAZWA_WOJEWODZTWA>) lub gmina: <nazwa> (powiat <nazwa>, woj. <NAZWA_WOJEWODZTWA>) jeśli nie
podano SYM >
===== KONIEC WERSJI V4 =====
},
"language" : < kod języka >,
"status" : {
  "code" : < status odpowiedzi >,
  "message" : < komunikat > :
}
}

```

### Opis kodów

procedure\_name->publication\_text->code

- ETA - opis przygotowany przez EPK i w trakcie akceptacji przez podmiot akceptujący
- EZA - opis przygotowany przez EPK i zaakceptowany przez podmiot akceptujący
- IZA - opis przygotowany przez podmiot akceptujący

procedure\_general->local\_jurisdiction->for\_entrepreneur

- ZASP\_NIE\_DOTYCZY - nie dotyczy

- ZASP\_MIE\_WYK - ze względu na miejsce wykonywania działalności
- ZASP\_SIEDZ - ze względu na siedzibę
- ZASP\_DOW\_URZ – dowolny urząd
- ZASP\_LOK\_PRZEDS - ze względu na lokalizację przedsięwzięcia
- ZASP\_MIE\_WYK\_SIEDZ - ze względu na miejsce wykonywania działalności lub siedzibę firmy
- ZASP\_SIE\_LUB\_MIE - ze względu na siedzibę lub miejsce zamieszkania
- ZASP\_LOK\_SPRZED - ze względu na lokalizację punktu sprzedaży
- ZASP\_PORT\_ZALAD - ze względu na port załadunku statku
- ZASP\_MIE\_POL\_OB - ze względu na miejsce położenia obiektu
- ZASP\_MIE\_WYK\_RO - ze względu na miejsce wykonywania robót
- ZASP\_MIE\_POL\_UPR - ze względu na miejsce położenia upraw
- ZASP\_POL\_PLA\_NAS - ze względu na położenie plantacji nasiennej
- ZASP\_PORT\_MAC - ze względu na port macierzysty statku
- ZASP\_MIE\_PROW\_POL - ze względu na miejsce prowadzenia połowu
- ZASP\_KORZ\_SRO - ze względu na miejsce korzystania ze środowiska
- ZASP\_ODP\_KOM - ze względu na miejsce powstawania odpadów komunalnych
- ZASP\_MIE\_ZAM - ze względu na miejsce zamieszkania
- ZASP\_MIE\_INW\_BUD - ze względu na miejsce inwestycji budowlanej
- ZASP\_MIE\_ROZ - ze względu na miejsce rozbiórki
- ZASP\_POL\_BUD - ze względu na położenie budynku
- ZASP\_LOK\_ZJA - ze względu na lokalizację zjazdu
- ZASP\_LOK\_ZAJ\_PDROG - ze względu na lokalizację zajmowanego pasa drogowego
- ZASP\_POL\_NIE - ze względu na położenie nieruchomości
- ZASP\_POL\_DRZE - ze względu na położenie drzewa
- ZASP\_LOK\_INST - ze względu na lokalizację instalacji
- ZASP\_MIE\_POL\_ZAB - ze względu na miejsce położenia zabytku
- ZASP\_MIE\_WYZBPR\_ODP - ze względu na miejsce wytwarzania, zbierania lub przetwarzania odpadów
- ZASP\_MIE\_IMP\_MAS - ze względu na miejsce imprezy masowej
- ZASP\_URZ\_SKR\_PRZE - ze względu na urząd skarbowy właściwy dla przedsiębiorcy (podatnika)
- ZASP\_MIE\_ODP\_CEL - ze względu na miejsce odprawy celnej
- ZASP\_MIE\_PRO\_KSI\_RAC - ze względu na miejsce prowadzenia ksiąg rachunkowych
- ZASP\_TRA\_PRZEJ – ze względu na trasę przejazdu
- ZASP\_SIEDZ\_URZ - ze względu na siedzibę urzędu, który wydał decyzję

procedure\_general->local\_jurisdiction->for\_citizen

- ZASO\_NIE\_DOTYCZY - nie dotyczy
- ZASO\_MIE\_ZAMEL - ze względu na miejsce zameldowania
- ZASO\_MIE\_ZAMIE - ze względu na miejsce zamieszkania
- ZASO\_DOW\_URZ – dowolny urząd
- ZASO\_LOK\_PRZEDS - ze względu na lokalizację przedsięwzięcia
- ZASO\_MIE\_ZAMEL\_ZAMIE - ze względu na miejsce zameldowania lub zamieszkania
- ZASO\_MIE\_WYK\_ZAW - ze względu na miejsce wykonywania zawodu

- ZASO\_MIE\_POB\_CUDZ - ze względu na miejsce pobytu cudzoziemca
- ZASO\_MIE\_WYK\_ROB - ze względu na miejsce wykonywania robót
- ZASO\_MIE\_POL\_UPR - ze względu na miejsce położenia upraw
- ZASO\_POL\_PLA\_NAS - ze względu na położenie plantacji nasiennej
- ZASO\_PORT\_MAC - ze względu na port macierzysty statku
- ZASO\_MIE\_PRO\_POL - ze względu na miejsce prowadzenia połowu
- ZASO\_KORZ\_SRO - ze względu na miejsce korzystania ze środowiska
- ZASO\_ODP\_KOM - ze względu na miejsce powstawania odpadów komunalnych
- ZASO\_MIE\_INW\_BUD - ze względu na miejsce inwestycji budowlanej
- ZASO\_MIE\_ROZ - ze względu na miejsce rozbiórki
- ZASO\_POL\_BUD - ze względu na położenie budynku
- ZASO\_LOK\_ZJA - ze względu na lokalizację zjazdu
- ZASO\_LOK\_ZAJ\_PDROG - ze względu na lokalizację zajmowanego pasa drogowego
- ZASO\_POL\_NIE - ze względu na położenie nieruchomości
- ZASO\_POL\_DRZE - ze względu na położenie drzewa
- ZASO\_LOK\_INST - ze względu na lokalizację instalacji
- ZASO\_MIE\_POL\_ZAB - ze względu na miejsce położenia zabytku
- ZASO\_MIE\_IMP\_MAS - ze względu na miejsce imprezy masowej
- ZASO\_MIE\_ODP\_CEL - ze względu na miejsce odprawy celnej
- ZASO\_MIE\_POL\_OBI - ze względu na miejsce położenia obiektu
- ZASO\_TRA\_PRZEJ – ze względu na trasę przejazdu
- ZASO\_SIEDZ\_URZ - ze względu na siedzibę urzędu, który wydał decyzję

procedure\_steps->details->documents->documents\_description->forms->form\_code:

- Or - Oryginał
- Uk -Uwierzytelniona kopia
- Ko - Kopia
- Ut - Uwierzytelnione tłumaczenie
- Dw - Przedstaw do wglądu
- Pe - Dokument elektroniczny

### Przykład wywołania

Pobranie szczegółów procedury

[http://\[adres\\_uslugi\]/ApiProcedures/getDetails?alias=10](http://[adres_uslugi]/ApiProcedures/getDetails?alias=10)

### Opis kodów

OK\_0000 - poprawna odpowiedź

ERR\_0004 - Niepoprawny typ pola

ERR\_0005 - Niepoprawny kod języka

ERR\_0006 - Przekroczona dopuszczalna długość pola

ERR\_0010 - Pole wykracza poza dopuszczalny zakres

ERR\_0013 - Element nie znajduje się na liście

ERR\_0014 - Pole nie jest ID

ERR\_0034 - Podana instytucja nie znajduje się w obszarze podanego zasięgu terytorialnego

ERR\_5108 - Parametry TERYT i SYM nie mogą wystąpić jednocześnie



ERR\_5109 - Podano nieprawidłowy format numeru SYM  
ERR\_5110 - Podany numer SYM nie istnieje

### Znaczenia „executing\_institution”

W ramach tej metody często pojawiają się elementy list typu „executing\_institution” (e-usługi, dokumenty, itp.). Są to elementy, które mogą wystąpić tylko w przypadku podania zasięgu terytorialnego lub konkretnego identyfikatora instytucji w parametrach wejściowych metody. Dane zawarte w odpowiedzi nie wynikają z opisu procedury, ale z dokumentów uszczegóławiających tworzonych przez daną instytucję.

## 5.6. getThematicCategories - Pobieranie katalogu tematycznego

Funkcja zwraca katalog tematyczne dla procedur

### Nazwa funkcji

**getThematicCategories**

### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |                  |   |
|-------------------------------|--------|-------------|-----------|------------------|---|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki          | opis  |
| lang                          | string | nie         | PL        | kod języka       | język tłumaczenia procedury   |
| epk_only                      | int    | nie         | 0         | na liście (0, 1) | Czy wyszukiwać tylko kategorie, których procedury, posiadają e-usługę typu EPK (zawierającą w adresie url fragment biznes.gov.pl) |

### Struktura danych wyjściowych

```
{
  "content" : {
    "ThematicCategories" : [ {
      "id" : < identyfikator katalogu tematycznego >,
      "name" : < nazwa katalogu tematycznego >
      ===== WERSJA V2 =====
      "url" : < url kategorii >
      ===== KONIEC WERSJI V2 =====
    } ]
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat > :
  }
}
```

### Przykład wywołania

[http://\[adres\\_usługi\]/ApiProcedures/getThematicCategories](http://[adres_usługi]/ApiProcedures/getThematicCategories)  
[http://\[adres\\_usługi\]/ApiProcedures/getThematicCategories?lang=en](http://[adres_usługi]/ApiProcedures/getThematicCategories?lang=en)



## Opis kodów

OK\_0000 - poprawna odpowiedź  
ERR\_0005 - Niepoprawny kod języka

## 5.7. getProfessionCategories - Pobieranie katalogu zawodów

Funkcja zwraca katalog zawodów powiązany z procedurami

### Nazwa funkcji

**getProfessionCategories**

### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |                  |  |
|-------------------------------|--------|-------------|-----------|------------------|--|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki          | opis   |
| lang                          | string | nie         | PL        | kod języka       | język tłumaczenia                            |
| with_assigned_procedures_only | int    | nie         | 0         | na liście (0, 1) | tylko powiązane z opublikowanymi procedurami |

### Struktura danych wyjściowych

```
{
  "content" : {
    "ProfessionCategories" : [ {
      "id" : < identyfikator katalogu zawodu >,
      "name" : < nazwa katalogu zawody >,
      "is_regulated" : < czy zawód regulowany >
    } ]
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

### Przykład wywołania

```
http://[adres_usługi]/ApiProcedures/getProfessionCategories
http://[adres_usługi]/ApiProcedures/getProfessionCategories?lang=en
http://[adres_usługi]/ApiProcedures/getProfessionCategories?with_assigned_procedures_only=1
```

## Opis kodów

OK\_0000 - poprawna odpowiedź  
ERR\_0005 - Niepoprawny kod języka  
ERR\_0013 - Element nie znajduje się na liście

## 5.8. getListForRespAuthority - Pobieranie listy procedur dla Instytucji Centralnych / Akceptujących

Metoda która umożliwi wyświetlanie wszystkich procedur znajdujących się na liście opisów procedur instytucji centralnej lub akceptującej w zakładce „Procedury przypisane” w widoku szczegółów instytucji na portalu.

### Nazwa funkcji

## getListForRespAuthority

### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |                                  |   |
|-------------------------------|--------|-------------|-----------|----------------------------------|---|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki                          | opis                                      |
| lang                          | string | nie         | PL        | kod języka                       | język tłumaczenia procedury               |
| institution_id                | int    | tak         |           |                                  | id instytucji centralnej lub akceptującej |
| page                          | int    | nie         | 1         | min wartość 1, max wartość 999   | numer strony wyszukiwania                 |
| limit                         | int    | nie         | 10        | min wartość 1, max wartość 99999 | liczba wyników na stronie                 |
| search                        | string | nie         |           | max 100 znaków                   | nazwa procedury                           |

### Struktura danych wyjściowych

```
{
  "content" : {
    "procedures" : [ {
      "id" : < id procedury >,
      "alias" < alias procedury >,
      "name" : < nazwa procedury >,
      "language" : < kod języka procedury >,
      "modification_date" : < data modyfikacji >,
      "status" : < status opublikowania procedury na portalu (Ni, Op, Pr) >
    } ],
    "rows_count" : < liczba rekordów ogółem >
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

### Opis słownika „status”

- Ni – Opis opublikowano na portalu, ale jest nieaktualny
- Op – Opis opublikowano na portalu
- Pr – W trakcie przygotowania

### Przykład wywołania

Pobranie listy procedur dla Ministerstwa Gospodarki, z domyślną liczbą wyników 10

[http://\[adres\\_usługi\]/ApiProcedures/getListForRespAuthority?institution\\_id=170](http://[adres_usługi]/ApiProcedures/getListForRespAuthority?institution_id=170)

### Opis kodów

OK\_0000 - poprawna odpowiedź  
OK\_0002 - poprawna odpowiedź, brak wyników

ERR\_0004 - Niepoprawny typ pola  
ERR\_0005 - Niepoprawny kod języka  
ERR\_0006 - Przekroczona dopuszczalna długość pola

ERR\_0010 - Pole wykracza poza dopuszczalny zakres  
ERR\_0014 - Pole nie posiada prawidłowej wartości dla ID  
ERR\_0031 - Instytucja o podanym id nie istnieje  
ERR\_0035 - Żądana strona wykracza poza listę wyników

## 6. ApiPnaInstitutions

### 6.1. getHintsByCity – pobieranie podpowiedzi po nazwie miejscowości

#### Opis

Metoda pobierająca listę instytucji po nazwie miejscowości

#### Nazwa funkcji

**getHintsByCity**

#### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |                  |                 |   |
|-------------------------------|--------|-------------|------------------|-----------------|---|
| nazwa zmiennej                | typ    | obowiązkowy | domyślna wartość | warunki         | opis  |
| category_ids                  | array  | Nie         |                  |                 | Lista kategorii instytucji  |
| city_prefix                   | string | Tak         |                  | Minimum 3 znaki | miejscowość danego urzędu   |
| limit                         | int    | Nie         | 10               |                 | Limit wyświetlania elementów na stronie   |
| procedure_id                  | string | Nie         |                  |                 | id procedury w kontekście której wyświetlana jest instytucja. Jeśli podany, wyniki zawężane są jedynie do instytucji realizujących daną procedurę |
| esp_only                      | int    | Nie         | 0                | 1 lub 0         | Jeżeli esp_only=1, funkcja zwraca tylko te instytucje, które mają niepusty adres Elektronicznej Skrzynki Podawczej.                               |

#### Struktura danych wyjściowych

```
{
  "content" : [ {
    "Institution" : {
      "id" : < id instytucji >,
      "name" : < nazwa instytucji >,
      "esp" : < nazwa skrzynki podawczej; tymczasowo nazwa epuap >,
      "epuap_id" : < identyfikator instytucji w epuap >
    }
  } ],
  "language" : < kod języka >,
  "status" : {
    "code" : < status_odpowiedzi >,
    "message" : < komunikat >
  }
}
```

#### Przykład wywołania

```
http://[adres_uslugi]/ApiPnaInstitutions/getHintsByCity?city_prefix=Pozna
```

#### Opis kodów

OK\_0000 poprawna odpowiedź

ERR\_0011 Pole wymagane

ERR\_0027 Pole ma mniejszą długość niż wymagana liczba znaków

ERR\_0004 Pole posiada niepoprawny typ

## 6.2. getListByLocation – pobranie instytucji na podstawie lokalizacji

### Opis

Odpowiada listą instytucji dla lokalizacji podanej przez użytkownika. Niezbędne jest ID procedury. Opcjonalne są: kategoria, oraz miasto lub kod pocztowy (miasto – rozumiane tutaj jako miasto wnioskującego, nie miasto w którym zlokalizowana jest instytucja). Parametr ulica i numer budynku są elementami dodatkowymi - filtracyjnymi. Gdy Procedura jest wykonywana przez wydział, funkcja zwróci dane adresowe wydziału oraz nazwę tego wydziału.

### Nazwa funkcji

**getListByLocation**

### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |                  |                    |  |
|-------------------------------|--------|-------------|------------------|--------------------|--|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślna wartość | warunki            | opis   |
| category_ids                  | array  | nie         |                  |                    | Kategoria instytucji   |
| city                          | string | nie         |                  |                    | miasto <u>wnioskującego</u> (zameldowania, zamieszkania, prowadzenia działalności ...), dla którego chce znaleźć urząd przynależny danego typu/typów |
| post_code                     | string | nie         |                  |                    | kod <u>wnioskującego</u> (zameldowania, zamieszkania, prowadzenia działalności ...), dla którego chce znaleźć urząd przynależny danego typu/typów    |
| street                        | string | nie         |                  |                    | ulica wnioskującego (zameldowania, zamieszkania, prowadzenia działalności ...), dla którego chce znaleźć urząd przynależny danego typu/typów         |
| number                        | string | nie         |                  |                    | numer budynku wnioskującego (zameldowania, zamieszkania, prowadzenia działalności ...), dla którego chce znaleźć urząd przynależny danego typu/typów |
| procedure_id                  | string | tak         |                  |                    | id procedury (obowiązkowe) w kontekście której wyświetlany jest urząd  |
| esp_only                      | int    | nie         | 0                | 1 lub 0            | Jeżeli esp_only=1, funkcja zwraca tylko te instytucje, które mają niepusty adres Elektronicznej Skrzynki Podawczej.                                  |
| institution_type_id           | Int    | nie         |                  |                    | Parametr ogranicza listę zwracanych instytucji do instytucji o typie, który odpowiada przekazanemu id.   |
| sym                           | Int    | Nie         |                  | Poprawny numer sym | Identyfikator sym miejscowości   |

## Struktura danych wyjściowych

```
{
  "content" : [ {
    "Institution" : {
      "id" : < id instytucji >,
      "city" : < nazwa miejscowości, kraj jeśli inny niż Polska >,
      "name" : < nazwa instytucji >,
      "department_name" : < nazwa wydziału instytucji lub null >
      "number" : < numer lokalu >,
      "post_code" : < kod pocztowy >,
      "street" : < ulica >,
      "voivodeship" : < nazwa województwa >,
      "voivodeship_teryt_id" : < identyfikator teryt >,
      "esp" : < skrzynka podawcza >,
      "epuap_id" : < identyfikator epuap >,
      =====WERSJA V2=====
      "email" : < adres e-mail z danych instytucji >,
      "phone" : < numery telefonów z danych instytucji >, (numery mogą występować po
przecinku)
      =====KONIEC WERSJI V2=====
    }
  } ],
  "language" : < kod języka >,
  "status" : {
    "code" : < status_odpowiedzi >,
    "message" : < komunikat >
  }
}
```

## Przykład wywołania

```
http://[adres_uslugi]/ApiPnaInstitutions/getListByLocation?procedure_id=538
http://[adres_uslugi]/ApiPnaInstitutions/getListByLocation?procedure_id=158&city=Poz
http://[adres_uslugi]/ApiPnaInstitutions/getListByLocation?procedure_id=158&city=Poz&instituti
on_type_id=53
```

## Opis kodów

OK\_0000 poprawna odpowiedź

ERR\_0011 Pole wymagane

ERR\_0004 Pole posiada niepoprawny typ

ERR\_0019 Wartość pola jest niepoprawnym ciągiem kodu pocztowego

## 7. ApiPnaAddress

### 7.1. getListByPostCode – lista miejscowości po kodzie pocztowym

#### Nazwa funkcji

**getListByPostCode**

#### Parametry wejściowe

| Parametry wysłane metodą GET |        |             |                  |                                |              |
|------------------------------|--------|-------------|------------------|--------------------------------|--------------|
| Nazwa zmiennej               | Typ    | obowiązkowy | domyślna wartość | warunki                        | opis         |
| post_code                    | string | tak         |                  | Kod pocztowy w formacie xx-xxx | Kod pocztowy |

#### Struktura danych wyjściowych

```
{
  "content" : {
    "voivodeship" : [ { < województwa >
      "name" : < nazwa >,
      "province" : [ { < powiaty >
        "name" : < nazwa >,
        "community" : [ { < gminy >
          "name" : < nazwa >,
          "city" : [ { < miasta >
            "name" : < nazwa >,
            "postbox" : < czy skrzynka pocztowa >
          } ]
        } ]
      } ]
    } ]
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status_odpowiedzi >,
    "message" : < komunikat >
  }
}
```

#### Przykład wywołania

```
http://[adres_usługi]/ApiPnaAddress/getListByPostCode?post_code=62-020
```

#### Opis kodów

OK\_0000 poprawna odpowiedź  
OK\_0002 brak rekordów do wyświetlenia

ERR\_0011 Pole wymagane  
ERR\_0004 Pole posiada niepoprawny typ  
ERR\_0019 Wartość pola jest niepoprawnym ciągiem kodu pocztowego

### 7.2. getStreetHints - pobranie podpowiedzi ulic

Metoda pobiera listę ulic pasujących do parametrów wejściowych

## Nazwa funkcji

**getStreetHints**

## Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |                  |                       |              |
|-------------------------------|--------|-------------|------------------|-----------------------|--------------|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślna wartość | warunki               | opis         |
| post_code                     | string | tak         |                  | Poprawny kod pocztowy | Kod pocztowy |
| city                          | string | nie         |                  |                       | miejsowość   |
| street_prefix                 | int    | tak         |                  | Min. 3 znaki          | ulica        |

## Struktura danych wyjściowych

```
{
  "content" : [ < lista nazw ulic > ],
  "language" : < kod języka >,
  "status" : {
    "code" : < status_odpowiedzi >,
    "message" : < komunikat >
  }
}
```

## Przykład wywołania

```
http://[adres_usługi]/ApiPnaAddress/getStreetHints?street_prefix=kosa
```

## Opis kodów

OK\_0000 poprawna odpowiedź  
OK\_0002 brak rekordów do wyświetlenia

ERR\_0011 Pole jest wymagane  
ERR\_0019 Wartość pola jest niepoprawnym ciągiem kodu pocztowego  
ERR\_0004 Pole posiada niepoprawny typ  
ERR\_0027 Pole ma mniejszą długość niż wymagana liczba znaków



## 8. ApiPublications

### 8.1. getDetails – Pobranie publikacji

Metoda zwraca obiekt publikacji na podstawie podanego aliasu, odpowiednie tłumaczenie zgodne z podanym parametrem **Lang** oraz poradnikami powiązаныmi z publikacją w języku zgodnym z parametrem **Lang**. W przypadku braku tłumaczenia domyślnie jeżeli istnieje to zwracana jest wersja w języku angielskim, a gdy nie istnieje to wersja w języku polskim.

#### Nazwa funkcji

getDetails

#### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |                   |  |
|-------------------------------|--------|-------------|-----------|-------------------|--|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki           | opis   |
| alias                         | string | tak         |           | max 255 znaków    | alias publikacji,<br>*TEST-xxx1-xxx2*yyy gdzie<br>xxx1 to id publikacji<br>xxx2 to timestamp<br>yyy to alias publikacji<br>pobiera publikację nawet jeżeli nie jest opublikowana |
| lang                          | string | nie         | PL        | poprawny kod lang | kod języka w standardzie ISO 639-1   |

#### Struktura danych wyjściowych

```
{
  "content" : {
    "Publication" : {
      "LifeSituations": [ { < powiązane sytuacje życiowe >
        "acronym" : < akronim >,
        "name" : < nazwa poradnika >,
        "language" : < język >
      } ],
      "alias" : < alias >,
      "text" : < treść >,
      "id" : < id >,
      "title" : < tytuł publikacji >,
      "created" : < data utworzenia >,
      "created_by" : < imię i nazwisko autora >,
      "modified" : < data ostatniej modyfikacji >,
      "modified_by" : < imię i nazwisko osoby modyfikującej >,
      "metadata_description" : < dodatkowy opis >,
      "metadata_keywords" : < słowa kluczowe >,
      "publication" : < wprowadzona data opublikowania >,
      "state" : < status publikacji >,
      "noindex" : < publikacja ukryta >
    }
  },
  "language" : < kod języka >,
  "status" : {
```

```

"code" : < status_odpowiedzi >,
"message" : < komunikat >
}
}

```

### Przykład wywołania

```
http://[adres_usługi]/ApiPublications/getDetails?alias=2103-testowa-publicacja&lang=de
```

### Opis kodów

OK\_0000 poprawna odpowiedź

ERR\_0005 Parametr 'lang' niezgodna ze standardem ISO 639-1

ERR\_0006 Niepoprawna długość

ERR\_0011 Pole jest wymagane

ERR\_0101 Nie odnaleziono publikacji

ERR\_0015 Użyto niepoprawnych znaków

## 8.2. getList – Pobranie listy publikacji

Pobiera listę opublikowanych publikacji, które nie są ukryte. Lista jest tworzona z podziałem na strony, na podstawie podanej frazy.

### Nazwa funkcji

getList

### Wersja

V2

### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |                   |                                    |
|-------------------------------|--------|-------------|-----------|-------------------|------------------------------------|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki           | opis                               |
| page                          | int    | nie         | 1         | max wartość 999   | numer strony                       |
| limit                         | int    | nie         | 10        | max wartość 9999  | liczba publikacji na stronie       |
| lang                          | string | nie         | PL        | poprawny kod lang | kod języka w standardzie ISO 639-1 |
| search                        | string | nie         |           | max 100 znaków    | wyszukiwana fraza                  |

### Struktura danych wyjściowych

```

{
  "content" : {
    "rows_count" : < liczba wszystkich wierszy >,
    "Publications" : [ {
      "Publication" : {
        "alias" : < alias >,
        "id" : < id >,
        "title" : < tytuł publikacji >,
        "lang" : < język publikacji >,
        "weight" : < waga sphinx >,
        "short_introduction" : < zawartość pola „Krótkie wprowadzenie” >
      }
    }
  ]
}

```

```
},  
"language" : < kod języka >,  
"status" : {  
  "code" : < status_odpowiedzi >,  
  "message" : < komunikat >  
}  
}
```

### *Przykład wywołania*

[http://\[adres\\_uslugi\]/ApiPublications/getList](http://[adres_uslugi]/ApiPublications/getList)

### *Opis kodów*

OK\_0000 poprawna odpowiedź

OK\_0002 brak rekordów do wyświetlenia

ERR\_0005 Parametr 'lang' niezgodny ze standardem ISO 639-1

ERR\_0006 Przekroczona dopuszczalna długość pola

## 9. ApiMail

### 9.1. sendMail – wysyłka wiadomości e-mail

Metoda umożliwia wysyłkę wiadomości email.

#### Nazwa funkcji

**sendMail**

#### Parametry wejściowe

| Parametry wysyłane metodą GET |         |             |           |   |   |
|-------------------------------|---------|-------------|-----------|---|---|
| Nazwa zmiennej                | Typ     | obowiązkowy | domyślnie | warunki   | opis  |
| email                         | string  | tak         |           | max 250 znaków  | Adres email odbiorcy  |
| reply_to                      | String  | nie         |           | max 250 znaków  | Adres email nadawcy   |
| template_code                 | string  | tak         |           | poprawny kod szablonu, max 250 znaków   | kod szablonu dostępny z poziomu zaplecza                                  |
| parameters                    | tablica | nie         |           | zgodność przesłanych elementów z szablonem  | Parametry zdefiniowane z szablonie są zastępowane przesłanymi wartościami |
| attachments                   | Tablica | Nie         |           | pojedynczy załącznik nie większy niż 50 MB, obsługiwane typy plików to rtf, txt, pdf, xls, xlsx, doc, docx, png, jpg, xml | Zakodowane w Base64 pliki   |

#### Struktura parametru parameters

```
array(  
  [nazwa parametru 1] = [wartość parametru 1],  
  [nazwa parametru 2] = [wartość parametru 2],  
  [nazwa parametru 3] = [wartość parametru 3],  
)
```

#### Struktura parametru attachments

```
array( (int) 0 =  
  array(  
    "data" = [treść załącznika w Base64],  
    "name" = [nazwa załącznika],  
    "mime" = [typ MIME pliku]  
  )  
)
```

#### Struktura danych wyjściowych

```
{
```

```

"content" : {
  "Mail" : {
    "message" : < treść >,
  }
}
"language" : < kod języka >,
"status" : {
  "code" : < status_odpowiedzi >,
  "message" : < komunikat >
}
}

```

### Przykład wywołania

[http://\[adres\\_uslugi\]/ApiMail/sendMail](http://[adres_uslugi]/ApiMail/sendMail)

W POST

Nagłówek: Content-Type: application/x-www-form-urlencoded

Body:

1. [email=test%40test.pl&template\\_code=wysylka](#)
2. email= test%40test.pl&template\_code=wysylka  
&parameters%5Btest%5D=test&parameters%5Btest2%5D=test2

### Opis kodów

OK\_0000 poprawna odpowiedź

ERR\_0001 Niepoprawna metoda protokołu. Wymagany POST.

ERR\_0003 Pole ma niepoprawny adres e-mail.

ERR\_0004 Pole posiada niepoprawny typ.

ERR\_0006 Niepoprawna długość.

ERR\_0011 Pole jest wymagane.

ERR\_0015 Użyto niepoprawnych znaków.

ERR\_0041 Pole ma niepoprawny format. Pole może zawierać tylko litery, cyfry, znak \_.

ERR\_0042 Zbyt duży rozmiar załącznika.

ERR\_0043 Niepoprawny typ załącznika.

ERR\_0044 Brak szablonu wiadomości o podanym kodzie.

ERR\_0045 Brak wymaganych parametrów dla szablonu wiadomości.

ERR\_0046 Podano nadmiarowe parametry dla szablonu.

ERR\_0047 Załącznik jest niekompletny.

## 9.2. getMessages – pobieranie wiadomości

Funkcja zwracająca wiadomości e-mail. Wykonanie tej funkcji jest możliwe tylko z poziomu środowiska QA.

### Nazwa funkcji

**getMessages**

### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |                        |   |
|-------------------------------|--------|-------------|-----------|------------------------|---|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki                | opis  |
| from                          | string | tak         |           | poprawna data z czasem | Data z czasem w formacie RRRR-MM-DD GG:MM:SS od której mają zostać zwrócone wiadomości, podanie czasu jest opcjonalne |
| subject                       | String | nie         |           |                        | Fragment tematu   |

|          |        |     |  |                       |                     |
|----------|--------|-----|--|-----------------------|---------------------|
| message  | string | nie |  |                       | Fragment wiadomości |
| to_email | string | nie |  | poprawny adres e-mail | E-mail adresata     |
| to_name  | string | nie |  |                       | Nazwisko adresata   |

### Struktura danych wyjściowych

```
{
  "content" : {
    "rows_count" : < liczba wszystkich wierszy >,
    "Messages" : [ {
      "Message" : {
        "subject" : < treść >,
        "message" : < treść >,
        "message_html" : < treść w formacie HTML >,
        "to_name" : < adresat >,
        "to_email" : < e-mail adresata >,
        "creation_date" : < data utworzenia >,
        "processing_date" : < data przetworzenia >,
        "is_processed" : < status przetworzenia >,
        "reply_to" : < odpowiedź do >,
        "processing_error_code" : < kod błędu przetwarzania >,
        "processing_error_description" : < opis błędu przetwarzania >
      }
    } ]
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

### Przykład wywołania

[http://\[adres usługi\]/ApiMail/getMessages?from=2016-07-05 12:10](http://[adres usługi]/ApiMail/getMessages?from=2016-07-05 12:10)

### Opis kodów

OK\_0000 Poprawna odpowiedź  
 OK\_0002 Poprawna odpowiedź, bez wyników  
 ERR\_0003 Pole ma niepoprawny adres e-mail.  
 ERR\_0011 Pole jest wymagane.  
 ERR\_0015 Użyto niepoprawnych znaków.  
 ERR\_0049 Format podanej daty z czasem jest nieprawidłowy  
 ERR\_5022 Nieprawidłowe środowisko

## 10. ApiUsers

### 10.1. getDeletedUsers – pobranie usuniętych użytkowników

Funkcja zwraca identyfikatory użytkowników którzy zostali usunięci z systemu

#### Nazwa funkcji

**getDeletedUsers**

#### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |   |         |         |
|-------------------------------|--------|-------------|---|---------|---------|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie                                 | warunki | opis    |
| date_from                     | string | nie         | 1 tydzień od data_do w postaci RRRR-MM-DD |         | Data od |
| date_to                       | String | nie         | Data aktualna w postaci RRR-MM-DD         |         | Data do |

#### Struktura danych wyjściowych

```
{
  "content" : {
    "users" : [ { <użytkownicy>
      "id": <identyfikator użytkownika>
    } ]
  }
  "language" : < kod języka >,
  "status" : {
    "code" : < status_odpowiedzi >,
    "message" : < komunikat >
  }
}
```

#### Przykład wywołania

```
http://[adres_usługi]/ApiUsers/getDeletedUsers
http://[adres_usługi]/ApiUsers/getDeletedUsers?date_from=2012-10-10
http://[adres_usługi]/ApiUsers/getDeletedUsers?date_from=2012-10-10&date_to=2015-10-10
```

#### Opis kodów

OK\_0000 poprawna odpowiedź

ERR\_0026 Wartość pola [nazwa pola] nie jest datą w postaci RRRR-MM-DD.  
ERR\_0048 Data [nazwa pola] nie może być większa od daty [nazwa pola].

## 11. ApiGuide

### 11.1. getDescription – pobranie opisu poradnika

Funkcja zwraca dane dotyczące poradnika

#### Nazwa funkcji

**getDescription**

#### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |                                  |                         |
|-------------------------------|--------|-------------|-----------|----------------------------------|-------------------------|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki                          | opis                    |
| id/acronym/alias              | string | tak         |           | Poprawny identyfikator poradnika | Identyfikator poradnika |
| language                      | string | nie         | pl        | Poprawny kod języka              | Wersja językowa pytania |

#### Struktura danych wyjściowych

```
{
  "content" : {
    "alias" : < aktualny, pełen alias poradnika (id-akronim) >,
    "id" : < identyfikator numeryczny poradnika >,
    "name" : < pełna nazwa poradnika >,
    "name_short" : < skrócona nazwa poradnika >,
    "keywords" : < słowa kluczowe >,
    "abstract" : < skrócony opis >,
    "description" : < opis HTML poradnika >,
    "estimated_execution_time" : < szacunkowy czas wykonania w mm:ss >
    =====WERSJA V2=====
    "creation_date" : < data dodania poradnika >,
    "modification_date" : < data modyfikacji poradnika >
    =====KONIEC WERSJI V2=====
    =====WERSJA V3=====
    "acts" : [ { < tablica aktów prawnych powiązanych z poradnikiem >
      "id" : < id aktu prawnego >,
      "name" : < nazwa aktu prawnego >,
      "link" : < link do aktu prawnego >,
      "type_code" : < kod aktu prawnego Us; Ro; Uch; Pu; In >,
      "type_description" : < opis kodu aktu prawnego (Ustawa; Rozporządzenie; Uchwała;
Przepisy unijne; Inny >,
      "language" : < język, w którym zwrócony został tytuł aktu prawnego >,
      "references" : [ { < lista nawiązań do innych aktów prawnych >
        "id" : < id aktu prawnego >,
        "name" : < nazwa aktu prawnego >,
        "link" : < link do aktu prawnego >,
        "type_code" : < kod aktu prawnego Us; Ro; Uch; Pu; In >,
        "type_description" : < opis kodu aktu prawnego (Ustawa; Rozporządzenie; Uchwała;
Przepisy unijne; Inny >,
        "language" : < język, w którym zwrócony został tytuł nawiązania do aktu prawnego
>,
      } ]
    } ],
    "visible_acts" : [ { < tablica aktów prawnych powiązanych z poradnikiem >
      "id" : < id aktu prawnego >,
      "name" : < nazwa aktu prawnego >,
      "link" : < link do aktu prawnego >,

```



```

        "type_code" : < kod aktu prawnego Us; Ro; Uch; Pu; In >,
        "type_description" : < opis kodu aktu prawnego (Ustawa; Rozporządzenie; Uchwała;
Przepisy unijne; Inny >,
        "language" : < język, w którym zwrócony został tytuł aktu prawnego >,
        "references" : [ { < lista nawiązań do innych aktów prawnych >
            "id" : < id aktu prawnego >,
            "name" : < nazwa aktu prawnego >,
            "link" : < link do aktu prawnego >,
            "type_code" : < kod aktu prawnego Us; Ro; Uch; Pu; In >,
            "type_description" : < opis kodu aktu prawnego (Ustawa; Rozporządzenie; Uchwała;
Przepisy unijne; Inny >,
            "language" : < język, w którym zwrócony został tytuł nawiązania do aktu prawnego
>,
        } ]
    } ],
}
=====KONIEC WERSJI V3=====

},
"language" : < kod języka odpowiedzi >,
"status_code" : < status odpowiedzi >,
"status_msg" : < komunikat statusu >
}

```

### Przykład wywołania

```

http://[adres_usługi]/ApiGuide/getDescription?acronym=121
http://[adres_usługi]/ApiGuide/getDescription?acronym=121-SWIADCZE_USLUGI_W_ZAKRESIE_AGROTURYSTYKI
http://[adres_usługi]/ApiGuide/getDescription?id=121
http://[adres_usługi]/ApiGuide/getDescription?id=121-SWIADCZE_USLUGI_W_ZAKRESIE_AGROTURYSTYKI
http://[adres_usługi]/ApiGuide/getDescription?alias=121
http://[adres_usługi]/ApiGuide/getDescription?alias=121-SWIADCZE_USLUGI_W_ZAKRESIE_AGROTURYSTYKI
http://[adres_usługi]/ApiGuide/getDescription?alias=121&lang=en
http://[adres_usługi]/ApiGuide/getDescription?alias=121&lang=de

```

### Opis kodów

status\_code:

OK\_0000 poprawna odpowiedź

ERR\_0005 Pole lang posiada niepoprawny kod języka.  
ERR\_0006 Pole alias ma większą długość niż 255 znaków.  
ERR\_0011 Pole alias jest wymagane.  
ERR\_5020 Nie odnaleziono poradnika o podanym ID

## 11.2. startGuide – uruchomienie poradnika

Funkcja rozpoczyna wykonywanie poradnika. Zwraca dane sesji, poradnika, pytania oraz możliwe odpowiedzi na pierwsze pytanie

### Nazwa funkcji

startGuide

### Parametry wejściowe

| Parametry wysyłane metodą GET |     |             |           |         |      |
|-------------------------------|-----|-------------|-----------|---------|------|
| Nazwa zmiennej                | Typ | obowiązkowy | domyślnie | warunki | opis |

|                  |        |     |    |                                  |  |
|------------------|--------|-----|----|----------------------------------|--|
| id/acronym/alias | string | tak |    | Poprawny identyfikator poradnika | Identyfikator poradnika  |
| language         | string | nie | pl | Poprawny kod języka              | Język poradnika  |
| user_id          | Int    | nie |    |                                  | Identyfikator użytkownika, służy do wiązania poradnika z zarejestrowanym użytkownikiem |

### Struktura danych wyjściowych

```
{
  content : {
    "id" : < identyfikator pierwszego pytania w poradniku >,
    "name" : < tekst pierwszego pytania >,
    =====WERSJA V3=====
    description : < opis HTML tego pytania >,
    =====KONIEC WERSJI V3=====
    =====WERSJA V2=====
    "additional_info" : < opis HTML poradnika >,
    "help" : < krótka informacja HTML ułatwiająca odpowiedź na pytanie >
    =====KONIEC WERSJI V2=====
    "type" : < typ pytania - radio/checkbox >,
    "hash" : < identyfikator uruchomionego procesu poradnika >,
    "progress" : < procent ukończenia poradnika >,
    "is_first" : < true/false - czy to pierwsze pytanie w poradniku >,
    "is_last" : < true/false - czy to jest ostatnie pytanie przed zakończeniem poradnika >,
    "guide_alias" : < aktualny, pełen alias poradnika (id-akronim) >,
    "guide_id" : < identyfikator numeryczny poradnika >,
    "guide_name" : < pełna nazwa poradnika >,
    "guide_name_short" : < skrócona nazwa poradnika >,
    "available_answers" : [ { < tablica możliwych odpowiedzi w pytaniu >
      "id" : < identyfikator odpowiedzi >,
      "name" : < treść odpowiedzi >,
      "type" : < typ odpowiedzi - simple/exclusive >
    } ],
    "last_answers_ids" : [ ] < tablica identyfikatorów udzielonych odpowiedzi >,
    "visited_questions" : [ { < lista odwiedzonych, aktywnych pytań >,
      "id" : < identyfikator pytania >,
      "name" : < tekst pytania >
    } ],
    "actions" : [ { < tablica dodatkowych informacji i czynności >
      "name" : < nazwa czynności >,
      "description" : < opis czynności >,
      "url" : < adres URL czynności >,
      "type" : < typ czynności >
    } ]
  },
  "language" : < kod języka odpowiedzi >,
  "status_code" : < status odpowiedzi >,
  "status_msg" : < komunikat statusu >
}
```

### Przykład wywołania

```
http://[adres_uslugi]/ApiGuide/StartGuide?acronym=121
http://[adres_uslugi]/ApiGuide/StartGuide?acronym=121-SWIADCZE_USLUGI_W_ZAKRESIE_AGROTURYSTYKI
http://[adres_uslugi]/ApiGuide/StartGuide?id=121
http://[adres_uslugi]/ApiGuide/StartGuide?id=121-SWIADCZE_USLUGI_W_ZAKRESIE_AGROTURYSTYKI
http://[adres_uslugi]/ApiGuide/StartGuide?alias=121
http://[adres_uslugi]/ApiGuide/StartGuide?alias=121-SWIADCZE_USLUGI_W_ZAKRESIE_AGROTURYSTYKI
http://[adres_uslugi]/ApiGuide/StartGuide?alias=121&lang=en
```

```
http://[adres_usługi]/ApiGuide/StartGuide?alias=121&lang=de
http://[adres_usługi]/ApiGuide/StartGuide?alias=121&lang=en&user_id=1
```

## Opis kodów

status\_code:

OK\_0000 poprawna odpowiedź

ERR\_0005 Pole lang posiada niepoprawny kod języka.

ERR\_0006 Pole alias ma większą długość niż 255 znaków.

ERR\_0011 Pole alias jest wymagane.

ERR\_0014 Pole user\_id nie posiada prawidłowej wartości dla ID.

ERR\_5016 Jeden lub więcej podprocesów wchodzących w skład poradnika nie jest opublikowany. Nie można rozpocząć poradnika

content-> actions->type:

- NORMAL - czynność
- INFO - informacja

### 11.3. getQuestion – pobranie konkretnego pytania wraz z odpowiedziami

Funkcja zwraca dane dotyczące poradnika, pytania, udzielonych odpowiedzi. Pobierając wcześniejsze pytanie nie są kasowane odpowiedzi, które nastąpiły po nim.

## Nazwa funkcji

getQuestion

## Parametry wejściowe

| Parametry wysłane metodą GET |        |             |   |   |                         |
|------------------------------|--------|-------------|---|---|-------------------------|
| Nazwa zmiennej               | Typ    | obowiązkowy | domyślnie   | warunki                                       | opis                    |
| hash                         | string | tak         |   | Poprawny hash sesji                           | Hash sesji              |
| question_id                  | int    | nie         | Ostatnie pytanie, na które nie udzielono odpowiedzi | Identyfikator pytania występujący w poradniku | Identyfikator pytania   |
| language                     | string | nie         | pl  | Poprawny kod języka                           | Wersja językowa pytania |

## Struktura danych wyjściowych

```
{
  "content": {
    "id": < identyfikator pytania w poradniku >,
    "name": < tekst pierwszego pytania >,
    "description": < opis HTML tego pytania >,
    "additional_info": < opis HTML poradnika >,
  }
}
```

```

"help" : < krótka informacja HTML ułatwiająca odpowiedź na pytanie >
=====KONIEC WERSJI V2=====
"type" : < typ pytania - radio/checkbox >,
"hash" : < identyfikator uruchomionego procesu poradnika >,
"progress" : < procent ukończenia poradnika >,
"is_first" : < true/false - czy to jest pierwsze pytanie w poradniku >,
"is_last" : < true/false - czy to jest ostatnie pytanie przed zakończeniem poradnika >,
"guide_alias" : < aktualny, pełen alias poradnika (id-akronim) >,
"guide_id" : < identyfikator numeryczny poradnika >,
"guide_name" : < pełna nazwa poradnika >,
"guide_name_short" : < skrócona nazwa poradnika >,
"available_answers" : [ < tablica możliwych odpowiedzi w pytaniu >
  "id" : < identyfikator odpowiedzi >,
  "name" : < treść odpowiedzi >,
  "type" : < typ odpowiedzi - simple/exclusive >
],
"last_answers_ids" : [ ] < tablica identyfikatorów udzielonych odpowiedzi >,
"visited_questions" : [ { < lista odwiedzonych, aktywnych pytań >
  "id" : < identyfikator pytania >,
  "name" : < tekst pytania >
} ],
"actions" : [ { < tablica dodatkowych informacji i czynności >
  "name" : < nazwa czynności >,
  "description" : < opis czynności >,
  "url" : < adres URL czynności >,
  "type" : < typ czynności >
} ]
},
"language" : < kod języka odpowiedzi >,
"status_code" : < status odpowiedzi >,
"status_msg" : < komunikat statusu >
}

```

### Przykład wywołania

```

http://[adres_usługi]/ApiGuide/getQuestion?hash=b5ddafb52ab3b7d1ac65c8ab2454a3cc8e338d00
http://[adres_usługi]/ApiGuide/getQuestion?hash=b5ddafb52ab3b7d1ac65c8ab2454a3cc8e338d00
&question_id=58
http://[adres_usługi]/ApiGuide/getQuestion?hash=b5ddafb52ab3b7d1ac65c8ab2454a3cc8e338d00&lang=
en
http://[adres_usługi]/ApiGuide/getQuestion?hash=b5ddafb52ab3b7d1ac65c8ab2454a3cc8e338d00&quest
ion_id=58&lang=en

```

### Opis kodów

status\_code:

OK\_0000 poprawna odpowiedź

ERR\_0005 Pole lang posiada niepoprawny kod języka.

ERR\_0011 Pole hash jest wymagane.

ERR\_0014 Pole question\_id nie posiada prawidłowej wartości dla ID.

ERR\_0022 Pole hash posiada nieprawidłową wartość hash.

ERR\_5025 Brak pytania o identyfikatorze [x] wśród tych, na które odpowiedziałeś do tej pory.

content-> actions->type:

- NORMAL - czynność
- INFO - informacja

## 11.4. setAnswer – odpowiadanie na pytanie

Funkcja zwraca dane dotyczące poradnika, następnego pytania, udzielonych odpowiedzi. Po ponownym udzieleniu odpowiedzi na pytanie znajdujące się wcześniej na stosie, kasowane są kolejne odpowiedzi następujące po nim.

### Nazwa funkcji

setAnswer

### Parametry wejściowe

| Parametry wysyłane metodą GET |                |             |   |                                  |  |
|-------------------------------|----------------|-------------|---|----------------------------------|--|
| Nazwa zmiennej                | Typ            | obowiązkowy | domyślnie   | warunki                          | opis   |
| hash                          | string         | tak         |   | Poprawny hash sesji              | Hash sesji   |
| answer_id                     | int,<br>string | tak         |   | Poprawna odpowiedź danym pytaniu | Identyfikator odpowiedzi, w przypadku wielowyboru identyfikatory odpowiedzi po przecinku |
| question_id                   | int            | nie         | Ostatnie pytanie, na które nie udzielono odpowiedzi | Identyfikator pytania poradnika  | Identyfikator pytania  |
| language                      | string         | nie         | pl  | Poprawny języka                  | Wersja językowa pytania  |

### Struktura danych wyjściowych

```
{
  content : {
    id : < identyfikator pytania w poradniku >,
    name : < tekst pierwszego pytania >,
    =====WERSJA V3=====
    description : < opis HTML tego pytania >,
    =====KONIEC WERSJI V3=====
    =====WERSJA V2=====
    "additional_info" : < opis HTML poradnika >,
    "help" : < krótka informacja HTML ułatwiająca odpowiedź na pytanie >
    =====KONIEC WERSJI V2=====
    type : < typ pytania - radio/checkbox >,
    hash : < identyfikator uruchomionego procesu poradnika >,
    progress : < procent ukończenia poradnika >,
    is_first : < true/false - czy to jest pierwsze pytanie w poradniku >,
    is_last : < true/false - czy to jest ostatnie pytanie przed zakończeniem poradnika >,
    guide_alias : < aktualny, pełen alias poradnika (id-akronim) >,
    guide_id : < identyfikator numeryczny poradnika >,
    guide_name : < pełna nazwa poradnika >,
    guide_name_short : < skrócona nazwa poradnika >,
    available_answers : [ { < tablica możliwych odpowiedzi w pytaniu >
      id : < identyfikator odpowiedzi >,
      name : < treść odpowiedzi >,
      type : < typ odpowiedzi - simple/exclusive >
    } ],
    last_answers_ids : [ ] < tablica identyfikatorów udzielonych odpowiedzi >,
  }
}
```

```

visited_questions : [ { < lista odwiedzonych, aktywnych pytań >
  id : < identyfikator pytania >,
  name : < tekst pytania >
} ],
actions : [ { < tablica dodatkowych informacji i czynności >
  name : < nazwa czynności >,
  description : < opis czynności >,
  url : < adres URL czynności >,
  type : < typ czynności >
} ]
},
language : < kod języka odpowiedzi >,
status_code : < status odpowiedzi >,
status_msg : < komunikat statusu >
}

```

### Przykład wywołania

```

http://[adres_usługi]/ApiGuide/setAnswer?hash=b5ddafb52ab3b7d1ac65c8ab2454a3cc8e338d00
&answer_id=202
http://[adres_usługi]/ApiGuide/setAnswer?hash=b5ddafb52ab3b7d1ac65c8ab2454a3cc8e338d00&answer_
id=202&question_id=58
http://[adres_usługi]/ApiGuide/setAnswer?hash=b5ddafb52ab3b7d1ac65c8ab2454a3cc8e338d00&answer_
id=202&lang=en
http://[adres_usługi]/ApiGuide/setAnswer?hash=b5ddafb52ab3b7d1ac65c8ab2454a3cc8e338d00&answer_
id=202&question_id=58&lang=en

```

### Opis kodów

status\_code:

OK\_0000 poprawna odpowiedź

ERR\_0005 Pole lang posiada niepoprawny kod języka.

ERR\_0011 Pole hash jest wymagane.

ERR\_0014 Pole question\_id nie posiada prawidłowej wartości dla ID.

ERR\_0022 Pole hash posiada nieprawidłową wartość hash.

ERR\_5004 Taka odpowiedź nie istnieje dla tego pytania.

ERR\_5025 Brak pytania o identyfikatorze [x] wśród tych, na które odpowiedziałeś do tej pory.

content-> actions->type:

- NORMAL - czynność
- INFO - informacja

## 11.5. getSummary – pobranie podsumowania

Funkcja zwraca dane dotyczące podsumowania wykonania poradnika

### Nazwa funkcji

getSummary

### Parametry wejściowe

Parametry wysyłane metodą GET

| Nazwa zmiennej | Typ    | obowiązkowy | domyślnie | warunki             | opis                         |
|----------------|--------|-------------|-----------|---------------------|------------------------------|
| hash           | string | tak         |           | Poprawny hash sesji | Hash sesji                   |
| language       | string | nie         | pl        | Poprawny kod języka | Wersja językowa podsumowania |

### Struktura danych wyjściowych

```
{
  content : {
    summary_title : < tytuł podsumowania >,
    summary_text : < tekst HTML podsumowania >,
    summary_actions : [ { < tablica dodatkowych informacji i czynności >
      name : < nazwa czynności >,
      description : < opis czynności >,
      url : < adres URL czynności >,
      type : < typ czynności >
    } ],
    provided_answers : [ { < tablica udzielonych odpowiedzi >
      question : < treść pytania >,
      answer : [ ] < lista treści odpowiedzi >
    } ],
    hash : < identyfikator uruchomionego procesu poradnika >,
    guide_alias : < aktualny, pełen alias poradnika (id-akronim) >,
    guide_id : < identyfikator numeryczny poradnika >,
    guide_name : < pełna nazwa poradnika >,
    guide_name_short : < skrócona nazwa poradnika >
  },
  language : < kod języka odpowiedzi >,
  status_code : < status odpowiedzi >,
  status_msg : < komunikat statusu >
}
```

### Przykład wywołania

```
http://[adres_usługi]/ApiGuide/getSummary?hash=b5ddafb52ab3b7d1ac65c8ab2454a3cc8e338d00
http://[adres_usługi]/ApiGuide/getSummary?hash=b5ddafb52ab3b7d1ac65c8ab2454a3cc8e338d00&lang=en
```

### Opis kodów

status\_code:

OK\_0000 poprawna odpowiedź

ERR\_0005 Pole lang posiada niepoprawny kod języka.

ERR\_0011 Pole hash jest wymagane.

ERR\_0022 Pole hash posiada nieprawidłową wartość hash.

content-> summary\_actions ->type:

- NORMAL - czynność
- INFO - informacja

## 11.6. getPDFGuide – pobranie podsumowania wykonania poradnika

Funkcja zwraca plik PDF z dotyczący podsumowania wykonania poradnika

## Nazwa funkcji

getPDFGuide

### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |                                 |                           |
|-------------------------------|--------|-------------|-----------|---------------------------------|---------------------------|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki                         | opis                      |
| hash                          | string | tak         |           | Poprawny hash zakończonej sesji | Hash zakończonej sesji    |
| language                      | string | nie         | pl        | Poprawny kod języka             | Wersja językowa pliku pdf |

### Struktura danych wyjściowych

< plik PDF >

### Przykład wywołania

```
http://[adres_usługi]/ApiGuide/getPDFGuide?hash=b5ddafb52ab3b7d1ac65c8ab2454a3cc8e338d00
http://[adres_usługi]/ApiGuide/getPDFGuide?hash=b5ddafb52ab3b7d1ac65c8ab2454a3cc8e338d00&lang=en
```

### Opis kodów

status\_code:

OK\_0000 poprawna odpowiedź

ERR\_0005 Pole lang posiada niepoprawny kod języka.

ERR\_0011 Pole hash jest wymagane.

ERR\_0022 Pole hash posiada nieprawidłową wartość hash.

ERR\_5006 Poradnik nie został zakończony.

## 11.7. getList – Pobranie listy poradników

Metoda zwraca listę wszystkich poradników posortowanych po nazwie poradnika. Możliwe jest podanie parametru **search**, **lang**, **limit**, **page**, **sort**. W przypadku pobierania listy w języku innym niż język polski, jeśli nie została znaleziona sytuacja życiowa w żądany języku zwracany jest poradnik w języku polskim

## Nazwa funkcji

getList

### Parametry wejściowe

| Parametry wysyłane metodą GET |        |             |           |                |  |
|-------------------------------|--------|-------------|-----------|----------------|--|
| Nazwa zmiennej                | Typ    | obowiązkowy | domyślnie | warunki        | opis   |
| search                        | string | nie         |           | max 100 znaków | umożliwia zawężenie zwracanej listy do elementów wyszukanych po nazwie lub po słowach kluczowych na podstawie przekazanego parametru |



|            |        |     |      |   |   |
|------------|--------|-----|------|---|---|
| lang       | string | nie | PL   | poprawny kod lang                             | kod języka w standardzie ISO 639-1  |
| limit      | int    | nie | 10   | max wartość 9999                              | ustawia ilość zwracanych jednorazowo elementów listy  |
| page       | int    | nie | 1    | max wartość 999                               | ustawia numer strony która ma zostać zwrócona, współpracuje z parametrem limit  |
| order      | string | nie | asc  | na liście (asc, desc), max 4 znaki            | ustawia kierunek sortowanie listy po nazwie sytuacji życiowej, przyjmuje dwie możliwe wartości: <i>asc</i> lub <i>desc</i>  |
| category   | array  | nie |      | Wartość z listy musi być zgodna ze słownikiem | Lista kategorii podana po przecinku z zakresu :<br>'ZAK' => 'Zakładam firmę',<br>'PRO' => 'Prowadzę firmę',<br>'ZAM' => 'Zamykam firmę',<br>'USL' => 'Tymczasowo świadczę usługi na terenie Polski' |
| importance | Int    | Nie |      | Z zakresu od 0 do 999                         | Określa ważność z przedziału od 0 do 999  |
| sort       | String | Nie | Name | Na liście, max 25 znaków                      | Wartość z listy name, category, importanc   |
| pkd        | string | nie |      |   | Poprawny kod PKD, z kropkami lub bez  |

### Struktura danych wyjściowych

```
{
  "content" : {
    "rows_count" : < liczba wszystkich wierszy >,
    "Guides" : [ {
      "Guide" : {
        "id" : < identyfikator sytuacji życiowej >,
        "acronym" : < akronim sytuacji życiowej >,
        "category" : < kod kategorii >,
        "importance" : < ważność poradnika >,
        "icon" : < link do ikonki sytuacji życiowych >,
        "name" : < nazwa sytuacji życiowej >,
        "language" : < język w którym został zwrócony element listy >,
        =====WERSJA V2=====
        "weight" : < waga >,
        =====KONIEC WERSJI V2=====
        "short_introduction" : < zawartość pola „Krótkie wprowadzenie” >
        "modification_date" : < data modyfikacji >,
      }
    } ]
  },
  "language" : < kod języka >,
  "status" : {
    "code" : < status odpowiedzi >,
    "message" : < komunikat >
  }
}
```

## Przykład wywołania

```
http://[adres_usługi]/ApiGuide/getList  
http://[adres_usługi]/ApiGuide/getList?limit=500  
http://[adres_usługi]/ApiGuide/getList?limit=500&search=otwieram
```

## Opis kodów

OK\_0000 - poprawna odpowiedź

ERR\_0004 - pole posiada [nazwa\_pola] niepoprawny typ. Wymagany typ to [typ]

ERR\_0005 - pole [nazwa\_pola] posiada niepoprawny kod języka

ERR\_0006 - pole [nazwa\_pola] ma większą długość niż [wymagana\_długość] znaków.

ERR\_0007 - pole [nazwa\_pola] większą wartość niż [maksymalna wartość].

ERR\_0007 - pole [nazwa\_pola] mniejsza wartość niż [maksymalna wartość].

ERR\_0013 - pole [nazwa\_pola] nie znajduje się na liście [wymagana lista parametrów].

ERR\_0024 - wartość pola z poza listy

ERR\_0025 - Wartość pola pkd jest niepoprawnym kodem PKD

ERR\_0035 - Żądana strona wykracza poza listę wyników

## Historia wersji (zmian) dokumentu

| Data       | Opis zmian  |
|------------|---|
| 20-12-2013 | Utworzenie dokumentu i opisanie aktualnych metod  |
| 10-01-2014 | Aktualizacja dokumentacji o nowe funkcje  |
| 13-01-2014 | Opisanie funkcji pobierającej szczegóły procedury   |
| 22-01-2014 | Aktualizacja dokumentacji dla metody getPkdWithHints  |
| 22-01-2014 | Opisanie nowej metody getProcedureCountByPkdid  |
| 24-01-2014 | Zmiana metody getProcedureCountByPkdid na getProcedureCountByPkdCode  |
| 27-01-2014 | getPKDWithHints zaktualizowano tabele kodów błędów.   |
| 28-01-2014 | Dodanie metody linkSessionWithUser związanej z sytuacjami życiowymi<br>Modyfikacja parametrów wyjściowych dla funkcji<br>getListToPerformElectronicProcedure  |
| 21-02-2014 | Dodanie department_id dla ApiProcedures dla metody getList,<br>Dodanie informacji o znaczniku w aktualnych czynnościach   |
| 03-03-2014 | Dodanie pól zwracanych przez metodę getDetails ApiProcedures:<br>name_abbreviation, keywords, keywords_additional<br>Dodanie rodzaju grupy do parametrów wejściowych metody getList ApiInstitutions:<br>UMG (urzędy miast i gmin)   |
| 06-03-2014 | Sortowanie listy procedur po dacie publikacji i dacie modyfikacji<br>Dodanie pola modification_date do wyników getList ApiProcedures  |
| 02-04-2014 | Zaktualizowanie kodów błędów dla voteOnLifeSituation. Dodanie informacji o parametrze Content-Type przy wysyłaniu metodą POST.  |
| 07-04-2014 | Dodanie metody revertToStage umożliwiającej cofnięcie sytuacji życiowej do wybranego etapu, aktualizacja parametrów wyjściowych dla getList i getDescription dla Sytuacji życiowych   |
| 10-04-2014 | Dodanie nowych pól do metod getDescription i getList dla API LifeSituation  |
| 25-04-2014 | Dodanie nowego kodu błędu do funkcji startLifeSituation   |
| 06-05-2014 | ApiLifesituation GetList dodanie możliwości sortowania po name,category, importance   |
| 14-05-2014 | ApiComons - zaktualizowano listę kodów odpowiedzi dla metody getPDF   |
| 28-05-2014 | ApiLifeSituations -> getDescription - aktualizacja listy kodów związanych z funkcjonalnością uruchomienia testowego sytuacji życiowej (ERR_5019, ERR_5020)  |
| 05-06-2014 | Aktualizacja dla ApiLifeSituation dla metody startLifeSituation z funkcjonalnością uruchomienia testowego sytuacji życiowej (ERR_5019, ERR_5020)  |
| 18-06-2014 | ApiProcedures GetDetails - dodanie nazwy instytucji do listy instytucji wykonujących procedurę,<br>Dla ApiInstitution dla metody getList dodano nowe pole voivodeship w odpowiedzi na zgłoszenie 0000091  |
| 01-07-2014 | Dla metody ApiProcedures getList poprawiono filtrację wyniku dla parametrów electronic_only i institution_id. Struktura odpowiedzi nie uległa zmianie   |
| 23-07-2014 | Dla ApiProcedur dla metody getDetail uzupełniono wzór odpowiedzi o elementy electronic_service_count, electronic_services   |
| 20-10-2014 | Zmieniono sposób odwoływania do procedur (poprzez alias). Zmieniono odpowiedź dla getDetails procedury  |
| 22-10-2014 | Aktualizacja getDetails procedury - dodatkowe opisy, zmiana update_date na confirmation_date, dodanie oprócz opisów kodów słownikowych również wartości kodów (code, code_description)  |
| 27-10-2014 | Aktualizacja getDetails instytucji - dodatkowe sekcje: InstitutionPkd oraz InstitutionTerritoryRange  |
| 28-10-2014 | Zmiana odpowiedzi API dla metody ApiLifeSituations->getPublication (zostały usunięte pola pub_tytuł, pub_alias, fulltext; pole alias zmieniło strukturę z postaci [string] na [int-string])<br>Dodanie bloku „forms” - form dokumentów w szczegółach dokumentu na kroku - ApiProcedures, getDetails |
| 05-11-2014 | Dokumentacja ApiPnaInstitutions - metody getHintsByCity oraz getListByLocation  |
| 06-11-2014 | Poprawiono strukturę statutory_duration na procedurze oraz na kroku, poprawiono kody typów aktów prawnych   |
| 06-11-2014 | Dokumentacji ApiPnaAddress metoda getListByPostCode,  |
| 07-11-2014 | Dokumentacja ApiPnaAddress metoda getStreetHints,   |

|            |   |
|------------|---|
|            | Poprawka dla opisu procedury->procedure_steps->details->deadline code/<br>code_description  |
| 14.11.2014 | Poprawa APIInstitutions->getDetails.<br>Zostało usunięte pole contact z element<br>InstitutionDepartments[DepartmentDetails]  |
| 19.11.2014 | Zmodyfikowano strukturę wyjściową dla ApiProcedures -> getDetails.<br>Zmieniono strukturę eservices dla procedury oraz dla kroków<br>Zmieniono strukturę opłat na kroku   |
| 02-12-2014 | Uzupełniono opisy dla ApiLifeSituations->startLifeSituation   |
| 05-12-2014 | Dodano informacje o planowanych zmianach w ramach sprintu 27 (19-12-2014) <ul style="list-style-type: none"> <li>• Dodano do struktury wyników: ApiLifeSituations: getPublication -&gt; „noindex”</li> <li>• Dodano do parametru wejściowego dodatkowy element słownikowy „INSTYTUCJA” ApiOpinions: saveOpinionForContent -&gt; „group”</li> <li>• Zmieniono strukturę wynikową dla ApiInstitutions: getDetails dodając: „city”, „number”, „post_code”, „street”, „general_eservice”</li> <li>• Zmieniono strukturę wynikową dla ApiInstitutions: getList dodając gałąź „departments”</li> <li>• Dodano parametr wejściowy : ApiInstitutions: getList -&gt; „with_departments”</li> <li>• Dodano do struktury wyników: ApiProcedures: getDetails -&gt; „is_expired”, „institutions_implementing_hint”</li> <li>• Dodano parametr wejściowy: ApiPnaInstitutions: getHintsByCity -&gt; „esp_only”</li> <li>• Dodano parametr wejściowy: ApiPnaInstitutions: getListByLocation -&gt; „esp_only”</li> <li>• Dodano nową grupę API: ApiPublications, a w niej metody getList i getDetails</li> </ul> |
| 12-12-2014 | Dodatek do Sprint 27 <ul style="list-style-type: none"> <li>• Zmieniono strukturę wynikową dla ApiInstitutions: getDetails dodając: „esp”, „epuap_id”</li> </ul>  |
| 18-12-2014 | Ujednolicono struktura aliasu testowego dla publikacji w ApiPublications i<br>ApiLifeSituations.  |
| 29-12-2014 | Uzupełniono ApiPnaInstitutions: getListByLocation o informacje o wydziale<br>instytucji.  |
| 16-01-2015 | <ul style="list-style-type: none"> <li>• Dodano do struktury wyników: ApiProcedures: getDetails -&gt; procedure_general pole „purpose” (cel procedury)</li> <li>• Dodano metodę ApiProcedures: getListForRespAuthority</li> </ul>   |
| 29-01-2015 | Dodano parametr wejściowy „range_teryt” do ApiProcedures: getList   |
| 06-02-2015 | Dodano ApiInstitutions: getDepartmentDetails  |
| 11-02-2015 | Usunięto szczegóły wydziału z ApiInstitutions: getDetails   |
| 13-02-2015 | Uzupełniono ApiLifeSituations->startLifeSituation i ApiLifeSituations->getDescription o dane katalogów tematycznych   |
| 17-02-2015 | Uzupełniono parametr „group” metody ApiInstitutions->getList o element<br>„POZOSTALE”   |
| 09-03-2015 | Uzupełniono ApiLifeSituations->getDescription o dane publikacji powiązanych   |
| 11-03-2015 | Dodano parameter epk_only do ApiProcedures/getList  |
| 17-03-2015 | Uzupełniono dane opłat o kod waluty w ApiProcedures->getDetails   |
| 19-03-2015 | Uzupełniono dane publikacji o powiązane poradniki w ApiPublications->getDetails   |
| 24-03-2015 | Dodano kod kontekstu (context_code) dla dokumentu do ApiProcedures->getDetails  |
| 25-03-2015 | Gdy podana instytucja nie znajduje się w obszarze podanego zasięgu terytorialnego<br>jest zgłaszany odpowiedni wyjątek dla ApiProcedures->getDetails  |
| 26-03-2015 | <ul style="list-style-type: none"> <li>• Dodanie danych instytucji akceptującej do ApiProcedures-&gt;getDetails</li> <li>• Wyszukiwanie w getListForRespAuthority również po instytucji akceptującej</li> </ul>   |
| 01-04-2015 | Usunięcie ApiLifeSituations->getPublication, zamienionej na ApiPublications->getDetails   |
| 09-04-2015 | <ul style="list-style-type: none"> <li>• Uporządkowanie opisu pól time_unit w ApiPrcedures-&gt;getDetails (wartości słownikowe po polsku)</li> <li>• Dodanie ApiInstitutions-&gt;getDetails-&gt;InstitutionDetails-&gt;responsible_procedures_count</li> </ul>  |
| 13-04-2015 | Dodanie informacji o procedurach powiązanych z procedurą poprzez domuent<br>wejściowy w ApiPrcedures->getDetails jako related_procedures dla każdego<br>dokumentu   |
| 17-04-2015 | Zmiana struktury ApiPrcedures->getDetails, umieszczenie pól publication_text i<br>accepting_institution poza obiektami.   |
| 30-04-2015 | Wydłużono pole wyszukiwania procedur do 300 znaków  |
| 06-05-2015 | Filtrowanie zawodów parametrem with_assigned_procedures_only  |

|            |   |
|------------|---|
| 19-05-2015 | Zmiana nazwy pola w odpowiedzi "RowCount na "rows_count" dla: ApiLifeSituations->getList, ApiPublications->getList  |
| 27-05-2015 | Dodanie właściwości "business_lifecycle_name" i "business_lifecycle" do obszaru "references" w ApiProcedures->getDetails  |
| 22-06-2015 | <ul style="list-style-type: none"> <li>• Usunięto metody ApiCommons: searchPKD, getProcedureCountByPkdCode, getPKDWithHints</li> <li>• Usunięto metodę ApiProcedures: getHintsQuestionPKD</li> <li>• Dodano metody ApiCommons: getPkdList, getPkdHints</li> <li>• Dodano kod błędu ERR_0035 do metod: ApiInstitutions-&gt;getList, ApiLifeSituations-&gt;getList, ApiProcedures-&gt;getElectronicList, ApiProcedures-&gt;getList, ApiProcedures-&gt;getListForRespAuthority</li> </ul>  |
| 01-07-2015 | Rozbudowano metody ApiInstitutions->getDetails i ApiInstitutions->getDepartmentDetails o dane dotyczące systemu kolejkowego   |
| 24-07-2015 | Modyfikacja akronimu sytuacji życiowych. Pole acronym może przyjmować wartości „id-akronim” lub „id” lub „akronim”. Wartość akronimu jest zwracana w odpowiedziach w postaci „id- akronim”  |
| 05-08-2015 | Modyfikacja danych wejściowych metody ApiOpinions/saveOpinionForContent   |
| 21-08-2015 | Dodanie pola institution_verification_date w szczegółach procedury  |
| 26-08-2015 | Dla ApiLifeSytuations/GetDescription dodano w odpowiedzi pole acronym   |
| 29-09-2015 | Dodanie kodu teryt do zasięgu terytorialnego wydziałów i instytucji.  |
| 30-09-2015 | Dodanie nowej metody ApiMail/sendEmail  |
| 07-10-2015 | <ul style="list-style-type: none"> <li>• Dodanie parametru PKD w liście Poradników</li> <li>• Dodanie anyone_has_epuap do szczegółów Procedury</li> </ul>   |
| 21-10-2015 | <ul style="list-style-type: none"> <li>• Dodanie pól related_procedures_count i related_life_situations_count do metody api ApiCommons/getPkdList</li> <li>• Dodanie powiązanych aktów prawnych do funkcji ApiLifeSituation/getDescription</li> </ul>   |
| 23-10-2015 | Dodanie sekcji pkd do metody api ApiLifeSituations/getDescription   |
| 04-11-2015 | Dodanie do sekcji "procedure_general" pola "description_code" zawierającego informacje na temat typu opisu procedury dla ApiProcedures/getDetails   |
| 18-11-2015 | Zmiana wartości pola has_fees w elemencie procedures_general gdy pole description_code = OPIS_SKROCONY dla ApiProcedures/getDetails   |
| 15-12-2015 | Dodanie nowej metody ApiUsers/getDeletedUsers   |
| 17-05-2016 | Nowe wartości słownikowe w polu procedure_general/local_jurisdiction  |
| 05-07-2016 | Dodanie nowej metody ApiMail/getMessages  |
| 08-07-2016 | Zmiany struktury odpowiedzi oraz opcjonalność czasu w ApiMail/getMessages   |
| 28-07-2016 | Zwracany błąd ERR_0021 w metodzie ApiOpinions/saveOpinionForContent, gdy przekazany parametr user_ip nie jest poprawnym adresem IP  |
| 16-09-2016 | Dodanie nowej wartości parametru procedure_execution->eservices->category w ApiProcedures/getDetails - „general_eservice”   |
| 04-11-2016 | Modyfikacja ustawowy czas realizacji procedury odpowiedzi dla Statutory_duration - usunięcie opcji "time" i "time_unit"   |
| 04-11-2016 | <ul style="list-style-type: none"> <li>• Wygaszenie pola przebieg alternatywny dla APIProcedures/getDetails</li> <li>• Wygaszenie pola rezultat procedury dla APIProcedures/getDetails</li> </ul>   |
| 18-11-2016 | Pole miejscowość w obiektach instytucji oraz wydziałów zawiera teraz dodatkowo kraj jeśli jest on inny niż Polska   |
| 23-11-2016 | <p>ApiProcedures/getDetails</p> <ul style="list-style-type: none"> <li>▪ procedure_steps-&gt;type_code - zmiana kodów (usunięcie kodu U)</li> <li>▪ procedure_steps-&gt;type_description - zmiana tekstów</li> <li>▪ procedure_steps-&gt;details-&gt;deadline-&gt;code - zmiana kodów (usunięcie kodu 'czas')</li> <li>▪ procedure_steps-&gt;details-&gt;deadline-&gt;code_description - zmiana tekstów</li> <li>▪ procedure_steps-&gt;details-&gt;deadline - usunięcie właściwości 'time' i 'time_unit'</li> <li>▪ procedure_steps-&gt;details-&gt;participants - właściwość nieobligatoryjna</li> </ul> |
|            | ApiProcedures/getListForRespAuthority - wartość „status” - usunięto kod Br  |
| 21-12-2016 | <ul style="list-style-type: none"> <li>• ApiProcedures/getDetails procedure_general-&gt; local_jurisdiction -&gt; for_entrepreneur - dodano: <ul style="list-style-type: none"> <li>○ ZASP_KORZ_SRO - ze względu na miejsce korzystania ze środowiska</li> <li>○ ZASP_ODP_KOM - ze względu na miejsce powstawania odpadów komunalnych</li> <li>○ ZASP_MIE_ZAM - ze względu na miejsce zamieszkania</li> </ul> </li> </ul>   |

|            |  |
|------------|--|
|            | <ul style="list-style-type: none"> <li>• ApiProcedures/getDetails<br/>procedurę_general-&gt; local_jurisdiction -&gt; for_citizen- dodano: <ul style="list-style-type: none"> <li>○ ZASO_KORZ_SRO - ze względu na miejsce korzystania ze środowiska</li> <li>○ ZASO_ODP_KOM - ze względu na miejsce powstawania odpadów komunalnych</li> </ul> </li> <li>• ApiProcedures/getDetails<br/>procedurę_general-&gt; statutory_duration – zmieniono:<br/>code_description -&gt;null<br/>description -&gt; tekst w zależności od pola 'code'</li> </ul>   |
| 11-01-2017 | <p>Modyfikacja i dodanie nowych pól dla metody getDetails dla APIProcedures.<br/>Dodanie pól:<br/>- procedure_execution-&gt;how_to_execute</p> <ul style="list-style-type: none"> <li>• by_institution</li> <li>• by_letter</li> <li>• by_other</li> <li>• by_internet</li> <li>• other_description</li> <li>• barrier_description</li> <li>• eservices</li> </ul> <p>przeniesienie gałęzi eservice do nowej gałęzi how_to_execute<br/>usunięcie pola z opisu kroku:<br/>"is_conditional_step" : &lt; czy krok jest warunkowy (true/false)&gt;,</p>  |
| 19-01-2017 | <p>Dodanie nowych pól dla metody getCurrentQuestion:</p> <pre>"content" : {   "question" : {     "answer_type" : &lt;typ odpowiedzi, jednokrotnego lub wielokrotnego wyboru: checkbox/radio&gt;</pre> <p>Oraz</p> <pre>"content" : {   "answers" : [ { &lt; tablicaodpowiedzi &gt;     "is_exclusive": &lt;dodatkowy parametr, który określa czy odpowiedź wyłącza możliwość wyboru innych (może wystąpić 1 na dane pytanie o typie 'checkbox')&gt;: false/true&gt;</pre> <p>Dodanie nowych błędów dla metody saveCurrentQuestion:</p> <p>ERR_5024 Dla pytania typu RADIO odpowiedź musi przyjmować jedną wartość<br/>ERR_5023 Zaznaczono w odpowiedzi wiele wartości, spośród których jedna posiada<br/>atrybut EXCLUSIVE</p> |
| 25-01-2017 | <p>Utworzono <b>wersję v2</b> metody getList:</p> <ol style="list-style-type: none"> <li>1. ApiPublication,</li> <li>2. ApiLifeSituation,</li> <li>3. ApiProcedures.</li> </ol> <p>Zmiana polega na dodaniu pola 'short_introduction'.<br/>Cel – ujednoczenie sposobu działania wyszukiwarek.</p>  |
| 09-02-2017 | <p>Zmodyfikowano wersję V3 metody getDetail dla APIProcedures<br/>Modyfikacja gałęzi <b>how_to_execute</b>:</p> <pre>how_to_execute : { &lt; w jaki sposób można zrealizowac usługę &gt;   options: [ &lt;lista opcji sposobów realizacji usługi – podzbiór INSTITUTION, MAIL, ONLINE&gt; ],   additional_description: &lt;dodatkowy opis który jeśli niepusty to można wykorzystać do opisanie innych sposobów załatwienia sprawy lub barier w powyższych sposobach&gt; eservices : []}</pre>   |
| 28-02-2017 | <ul style="list-style-type: none"> <li>• Zmodyfikowano wersję V3 metody getDetails dla ApiProcedures<br/>Gałąź <b>procedure_execution</b>:<br/>Dodano pole 'room_person'</li> </ul>  |



|            |  |
|------------|--|
|            | <ul style="list-style-type: none"> <li>Zmodyfikowano metodę getDetails dla ApiInstitutions</li> </ul> <p>Gałąź <b>Institution</b>:</p> <p>Dodano pole 'executive_name'</p> <ul style="list-style-type: none"> <li>Zmodyfikowano metodę getList dla ApiInstitutions</li> </ul> <p>Dodano pole 'executive_name'</p>  |
| 01-06-2017 | Dodano nowe właściwości miejscowe w sekcji Zasięg realizowania usługi ze względu na właściwość miejscową – ApiProcedures/getDetails<br>for_entrepreneur: ZASP_MIE_INW_BUD; ZASP_MIE_ROZ; ZASP_POL_BUD; ZASP_LOK_ZJA; ZASP_LOK_ZAJ_PDROG; ZASP_POL_NIE; ZASP_POL_DRZE; ZASP_LOK_INST; ZASP_MIE_POL_ZAB; ZASP_MIE_WYZBPR_ODP; ZASP_MIE_IMP_MAS; ZASP_URZ_SKR_PRZE; ZASP_MIE_ODP_CEL<br>for_citizen: ZASO_MIE_INW_BUD; ZASO_MIE_ROZ; ZASO_POL_BUD; ZASO_LOK_ZJA; ZASO_LOK_ZAJ_PDROG; ZASO_POL_NIE; ZASO_POL_DRZE; ZASO_LOK_INST; ZASO_MIE_POL_ZAB; ZASO_MIE_IMP_MAS |
| 07-06-2017 | ApiProcedures/getDetails – w przypadku wybrania typu specjalnego instytucji realizującej dla usługi<br>"procedure_execution_type" = CENTRAL,<br>"institutions_implementing_in_range" = null<br>"institutions_implementing_all":{<br>"units": null LUB [<lista instytucji realizujących>]<br>"types": null},  |
| 08-06-2017 | ApiProcedures/getSYM - pozwala na wyszukanie prawidłowej miejscowości w oparciu o bazę SIMC  |
| 13-06-2017 | ApiProcedures/getDetails – dodano wersję V4 metody, która pobiera dodatkowy opcjonalny parametr „sym” i na podstawie terytu lub sym zwraca wartość „localization_name”. Metoda zwraca dodatkowo 3 rodzaje błędów związanych z parametrem „sym”   |
| 01-08-2017 | Dodano nowe właściwości miejscowe w sekcji Zasięg realizowania usługi ze względu na właściwość miejscową – ApiProcedures/getDetails<br>for_entrepreneur: ZASP_MIE_PRO_KSI_RAC<br>for_citizen: ZASO_MIE_POL_OBI; ZASO_MIE_ODP_CEL   |
| 17-08-2017 | Zmodyfikowano kody błędów dla ApiCommons/getSYM  |
| 11-10-2017 | Dodano nowe właściwości miejscowe w sekcji Zasięg realizowania usługi ze względu na właściwość miejscową – ApiProcedures/getDetails<br>for_entrepreneur: ZASP_TRA_PRZEJ<br>for_citizen: ZASO_TRA_PRZEJ   |
| 19-10-2017 | Dodano opis do form zwracanych dokumentów - ApiProcedures/getDetails<br>Or;Uk;Ko;Ut;Dw;Pe  |
| 15-02-2018 | Dodanie dokumentacji ApiGuide  |
| 23-02-2018 | Utworzono wersję V5 metody ApiProcedures/getDetails<br>Nowa wersja zawiera pole 'expiration_date' zwracająca datę oflagowania opisu usługi jako nieaktualny.   |
| 10-03-2018 | Utworzono wersję 2 metody getPKDDetails w ApiCommon  |
| 11-03-2018 | Utworzono wersję 2 metody getPKD w ApiCommon   |
| 12-03-2018 | Utworzenie wersji 2 metody getDetailsV2 w ApiInstitutions  |
| 13-03-2018 | Dla metody getHints w ApiProcedures dodano obsługę parametrów wejściowych institution_id oraz department_id  |
| 14-03-2018 | Dla metody getPKDDetails dodano nowe pola: related_procedures i related_life_situations zwracające odpowiednio usługi i poradniki powiązane z PKD  |
| 21-03-2018 | Utworzono wersję 2 metody getListByLocation w ApiPnaInstitutions   |
| 22-03-2018 | Utworzono wersję 2 metody getDescription w ApiGuide  |
| 22-03-2018 | Utworzono wersję 3 metody getList w ApiLifeSituations  |
| 25-03-2018 | Utworzono wersję 2 metod startGuide, getQuestion, setAnswer w ApiGuide   |
| 26-03-2018 | Dodano parameter central_only do ApiProcedures/getList   |
| 27-03-2018 | Utworzono wersję 3 metod startGuide, getQuestion, setAnswer w ApiGuide<br>Metoda ta usuwa pole description ze zwracanej przez API odpowiedzi.  |
| 29-03-2018 | Utworzono metodę getList dla ApiGuide  |
| 18-04-2018 | Utworzono wersję 4 metody getList w ApiLifeSituations  |
| 18-04-2018 | Utworzono wersję 2 metody getList w ApiGuide   |
| 21-04-2018 | Utworzono wersję 2 metody getThematicCategories w ApiProcedures. Dodanie nowej flagi epk_only dla getThematicCategories.   |
| 23-04-2018 | Utworzono wersję 3 metody getList w ApiProcedures  |
| 05-06-2018 | Utworzono wersję 6 metody getDetails w ApiProcedures   |

|            |  |
|------------|--|
| 09-08-2018 | Dodano możliwość stosowania encji URLowych w parametrach do zapytań API  |
| 03-09-2018 | Dodano nowe właściwości miejscowe w sekcji Zasięg realizowania usługi ze względu na właściwość miejscową – ApiProcedures/getDetails<br>for_entrepreneur: ZASP_SIEDZ_URZ<br>for_citizen: ZASO_SIEDZ_URZ |
| 11-09-2018 | Dodano w<br>ApiLifeSituations/getDescription v2 – visible_acts<br>ApiGuide/getDescription v3 – acts, visible_acts  |
| 09-10-2018 | Dodano pole 'eservice' w ApiProcedures/getList v4  |
| 11-10-2018 | Dodano filtr esp w ApiInstitutions/getDetails  |
| 16-10-2018 | Dodano objaśnienie dot. Pola 'eservice' w ApiProcedures/getList  |
| 23-10-2018 | Dodanie parametru sym w getListByLocation  |