

InfoPortugal POIS Service API

Author: Nuno Mariz
Version: 3.0
Copyright: InfoPortugal S.A.
Date: 7 Jan 2011



Versions:

Author: André Lima
Version: 3.2
Copyright: InfoPortugal S.A.
Date: ???

Author: Diogo Marques
Version: 3.3
Copyright: InfoPortugal S.A.
Date: 22 Abril 2013

Contents

1	Introduction	4
2	Points Of Interest search	5
2.1	Request	5
2.2	Parameters	5
2.3	Response	7
2.3.1	GeoJSON	7
3	Points Of Interest detail	9
3.1	Request	9
3.2	Parameters	9
3.3	Response	10
3.3.1	GeoJSON	10
4	Types of the Points Of Interest	12
4.1	Request	12
4.2	Parameters	12
4.3	Response	13
4.3.1	JSON	13
5	Subtypes of the Points Of Interest	14
5.1	Request	14
5.2	Parameters	14
5.3	Response	15
5.3.1	JSON	15
6	Classifications of the Points Of Interest	16
6.1	Request	16
6.2	Parameters	16
6.3	Response	17
6.3.1	JSON	17
7	Categories of the Points Of Interest	18
7.1	Request	18
7.2	Parameters	18
7.3	Response	19
7.3.1	JSON	19
8	Regions of the Points Of Interest	20
8.1	Request	20
8.2	Parameters	20
8.3	Response	21

8.3.1	JSON	21
9	Filters	23
9.1	Request	23
9.2	Parameters	23
9.3	Response	24
9.3.1	JSON	24

1 Introduction

This document describes the InfoPortugal POIS Service API.

2 Points Of Interest search

Service for searching Points Of Interest

2.1 Request

- (GET) /pois.json

2.2 Parameters

- lon: <decimal> Longitude
- lat: <decimal> Latitude
- distance: <integer> Distance in meters
- bbox: <decimal>,<decimal>,<decimal>,<decimal> Bounding box
- name: <string> Name
- type <integer> Type ID or list of ID's(comma separated)
- subtype <integer> Subtype ID or list of ID's(comma separated)
- classification <integer> Classification ID or list of ID's(comma separated)
- category <integer> Category ID or list of ID's(comma separated)
- region <integer> Region ID or list of ID's(comma separated)
- district <integer> District ID or list of ID's(comma separated)
- municipality <integer> Municipality ID or list of ID's(comma separated)
- ambiances: <integer> Integer or list of integers(comma separated)
- decorations: <integer> Integer or list of integers(comma separated)
- cuisines: <integer> Integer or list of integers(comma separated)
- payments: <integer> Integer or list of integers(comma separated)
- parks: <integer> Integer or list of integers(comma separated)
- sights: <integer> Integer or list of integers(comma separated)
- sports: <integer> Integer or list of integers(comma separated)
- services: <integer> Integer or list of integers(comma separated)
- general-services: <integer> Integer or list of integers(comma separated)
- room-services: <integer> Integer or list of integers(comma separated)
- children: <Integer> [0=false|1=true]
- smokers: <Integer> [0=false|1=true]
- animals: <Integer> [0=false|1=true]
- accessible-disabled: <Integer> [0=false|1=true]

2 Points Of Interest search

- limit: <integer> Limit of rows - Max=100
- lang: <string> Language [pt|en]
- callback: <string> JSONP Callback - Only available if the server supports
- api_key: <string> API key (**required**)

Important

The request must provide the lon, lat or the bbox or the name or the region.

2.3 Response

Note

The `distance` in the `properties` will return `null` if the request is for the `bbox`.

2.3.1 GeoJSON

GeoJSON is a format for encoding a variety of geographic data structures. A GeoJSON object may represent a geometry, a feature, or a collection of features. GeoJSON supports the following geometry types: Point, LineString, Polygon, MultiPoint, MultiLineString, MultiPolygon, and GeometryCollection. Features in GeoJSON contain a geometry object and additional properties, and a feature collection represents a list of features.

The service will return a **FeatureCollection**.

```
{
  "type": "FeatureCollection",
  "features": [{
    "geometry": {
      "type": "Point",
      "coordinates": [-8.5342910046, 37.1182789940]
    },
    "type": "Feature",
    "properties": {
      "category": "5 Stars",
      "website": "http://www.solveverde.pt/",
      "fax": "282402099",
      "name": "Algarve Casino",
      "classification": "Hotels",
      "phone": "282402000",
      "postal_locality": "081103",
      "review": {
        "Ratings": "5 Stars",
        "Minimum price of a double room": "99.00",
        "Accessible to the disabled": true,
        "Cuisine": "International; Traditional Portuguese; Vegetarian",
        "Sports": "Adventure sports; Water sports",
        "Number of private rooms": "9",
        "Ambience": "Familiar; Business; Romantic",
        "Internet Types": "Wireless",
        "Location": "Seaside",
        "Decorations": "Classic; Beach; Traditional",
        "Capacity of the meeting rooms": "1044",
        "Includes breakfast": true,
        "Has restaurant": true,
        "Number of beds": "418",
        "Groups": "Accepted",
        "Maximum price of a double room": "298.00",
        "Animals": "Allowed",

```

2.3 Response

```

        "Capacity of private rooms": "4",
        "Minimum price of a single room": "79.00",
        "Car park": "Easy external parking; Private parking",
        "Payments": "American Express; Check; Mastercard; ATM; Visa",
        "Maximum price of a single room": "258.00",
        "Number of rooms": "208",
        "Number of meeting rooms": "9",
        "Price of breakfast": "18.00",
        "Sights": "Town; Sea; Swimming Pool",
        "Number of suites": "16",
        "Bookings": "Recommended",
        "Internet Options": "Free",
        "Children": "Child-Friendly",
        "Smokers": "Smoking areas"
    },
    "distance": null,
    "photos": [],
    "email": "hotelalgarve@solverde.pt",
    "subtype": "Hospitality",
    "postal_code": "8500-802",
    "address": "Avenida Tomás Cabreira",
    "address_number": null,
    "type": "Accommodation",
    "id": "H0.611",
    "review_fields": [
        "Cuisine",
        "Minimum price of a double room",
        "Accessible to the disabled"
    ]
},
}],
"result": {
    "totalPages": 91,
    "itemsPerPage": 1,
    "totalResults": 91,
    "page": 1
}
}

```


3 Points Of Interest detail

Service for Points Of Interest detail

3.1 Request

- (GET) /pois.detail.json

3.2 Parameters

- id: <string> Poi ID (**required**)
- callback: <string> JSONP Callback - Only available if the server supports
- lang: <string> Language [pt|en]
- api_key: <string> API key (**required**)

3.3 Response

3.3.1 GeoJSON

GeoJSON is a format for encoding a variety of geographic data structures. A GeoJSON object may represent a geometry, a feature, or a collection of features. GeoJSON supports the following geometry types: Point, LineString, Polygon, MultiPoint, MultiLineString, MultiPolygon, and GeometryCollection. Features in GeoJSON contain a geometry object and additional properties, and a feature collection represents a list of features.

The service will return a **Feature**.

```
{
  "geometry": {
    "type": "Point",
    "coordinates": [-8.5342910046, 37.1182789940]
  },
  "type": "Feature",
  "properties": {
    "category": "5 Stars",
    "website": "http://www.solveverde.pt/",
    "fax": "282402099",
    "name": "Algarve Casino",
    "classification": "Hotels",
    "phone": "282402000",
    "postal_locality": "081103",
    "review": {
      "Ratings": "5 Stars",
      "Minimum price of a double room": "99.00",
      "Accessible to the disabled": true,
      "Cuisine": "International; Traditional Portuguese; Vegetarian",
      "Sports": "Adventure sports; Water sports",
      "Number of private rooms": "9",
      "Ambience": "Familiar; Business; Romantic",
      "Internet Types": "Wireless",
      "Location": "Seaside",
      "Decorations": "Classic; Beach; Traditional",
      "Capacity of the meeting rooms": "1044",
      "Includes breakfast": true,
      "Has restaurant": true,
      "Number of beds": "418",
      "Groups": "Accepted",
      "Maximum price of a double room": "298.00",
      "Animals": "Allowed",
      "Capacity of private rooms": "4",
      "Minimum price of a single room": "79.00",
      "Car park": "Easy external parking; Private parking",
      "Payments": "American Express; Check; Mastercard; ATM; Visa",
      "Maximum price of a single room": "258.00",
      "Number of rooms": "208",
      "Number of meeting rooms": "9",
      "Price of breakfast": "18.00",
      "Sights": "Town; Sea; Swimming Pool",
      "Number of suites": "16",
    }
  }
}
```

3.3 Response

```
    "Bookings": "Recommended",
    "Internet Options": "Free",
    "Children": "Child-Friendly",
    "Smokers": "Smoking areas"
  },
  "photos": [],
  "email": "hotelalgarve@solverde.pt",
  "subtype": "Hospitality",
  "postal_code": "8500-802",
  "address": "Avenida Tomás Cabreira",
  "address_number": null,
  "type": "Accommodation",
  "id": "H0.611",
  "review_fields": [
    "Cuisine",
    "Minimum price of a double room",
    "Accessible to the disabled"
  ]
}
```

4 Types of the Points Of Interest

Service for retrieving the types of the Points Of Interest

4.1 Request

- (GET) /pois.types.json

4.2 Parameters

- api_key: <string> API key (**required**)

4.3 Response

4.3.1 JSON

JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate.

The service will return in a JSON (JavaScript Object Notation) format.

```
{
  "types": [{
    "id": 9,
    "name": "Accommodation"
  },
  {
    "id": 8,
    "name": "Culture"
  },
  {
    "id": 7,
    "name": "Education"
  },
  {
    "id": 4,
    "name": "Health & Wellness"
  },
  {
    "id": 5,
    "name": "Restaurants and nightlife"
  },
  {
    "id": 2,
    "name": "Services"
  },
  {
    "id": 3,
    "name": "Shopping"
  },
  {
    "id": 6,
    "name": "Sports"
  },
  {
    "id": 1,
    "name": "Visit"
  }
]}
```

5 Subtypes of the Points Of Interest

Service for retrieving the subtypes of the Points Of Interest

5.1 Request

- (GET) `/pois.subtypes.json`

5.2 Parameters

- `type <integer>` Type ID (**required**)
- `api_key: <string>` API key (**required**)

5.3 Response

5.3.1 JSON

JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate.

The service will return in a JSON (JavaScript Object Notation) format.

```
{
  "subtypes": [{
    "id": 45,
    "name": "Family-run hotels"
  },
  {
    "id": 44,
    "name": "Hospitality"
  },
  {
    "id": 47,
    "name": "INATEL"
  },
  {
    "id": 42,
    "name": "Local accommodation"
  },
  {
    "id": 59,
    "name": "Resorts"
  },
  {
    "id": 43,
    "name": "Rural tourism"
  },
  {
    "id": 41,
    "name": "Tourist apartments"
  },
  {
    "id": 46,
    "name": "Tourist villages"
  },
  {
    "id": 48,
    "name": "Youth hostels"
  }
]
```

6 Classifications of the Points Of Interest

Service for retrieving the classifications of the Points Of Interest

6.1 Request

- (GET) /pois.classifications.json

6.2 Parameters

- subtype <integer> Subtype ID (**required**)
- api_key: <string> API key (**required**)

6.3 Response

6.3.1 JSON

JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate.

The service will return in a JSON (JavaScript Object Notation) format.

```
{
  "classifications": [{
    "id": 80,
    "name": "Apartment hotel"
  },
  {
    "id": 76,
    "name": "Hotels"
  },
  {
    "id": 81,
    "name": "Pousadas de Portugal"
  },
  {
    "id": 113,
    "name": "Private"
  },
  {
    "id": 112,
    "name": "Public"
  }
}]
}
```

7 Categories of the Points Of Interest

Service for retrieving the categories of the Points Of Interest

7.1 Request

- (GET) /pois.categories.json

7.2 Parameters

- classification <integer> Classification ID (**required**)
- api_key: <string> API key (**required**)

7.3 Response

7.3.1 JSON

JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate.

The service will return in a JSON (JavaScript Object Notation) format.

```
{
  "categories": [{
    "id": 9,
    "name": "1 Star"
  },
  {
    "id": 10,
    "name": "1st Category"
  },
  {
    "id": 11,
    "name": "3rd Category"
  },
  {
    "id": 1,
    "name": "4 Stars"
  },
  {
    "id": 5,
    "name": "5 Stars"
  },
  {
    "id": 15,
    "name": "Agrotourism"
  },
  {
    "id": 7,
    "name": "Charm Inn"
  },
  {
    "id": 8,
    "name": "Cottages"
  }
}]
}
```

8 Regions of the Points Of Interest

Service for retrieving the regions of the Points Of Interest

8.1 Request

- (GET) `/pois.regions.json`

8.2 Parameters

- `api_key: <string>` API key (**required**)

8.3 Response

8.3.1 JSON

JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate.

The service will return in a JSON (JavaScript Object Notation) format.

```
{
  "regions": [{
    "id": 1,
    "name": "Lisboa e Vale do Tejo"
  },
  {
    "id": 2,
    "name": "Centro"
  },
  {
    "id": 3,
    "name": "Alentejo"
  },
  {
    "id": 4,
    "name": "Algarve"
  },
  {
    "id": 5,
    "name": "Norte"
  },
  {
    "id": 6,
    "name": "Açores"
  },
  {
    "id": 7,
    "name": "Madeira"
  },
  {
    "id": 8,
    "name": "Douro"
  },
  {
    "id": 9,
    "name": "Serra da Estrela"
  },
  {
    "id": 10,
    "name": "Leiria - Fátima"
  },
  {
    "id": 11,
    "name": "Oeste"
  },
  },
}
```

8.3 Response

```
{
  "id": 12,
  "name": "Alqueva"
},
{
  "id": 13,
  "name": "Litoral Alentejano"
}]
}
```

9 Filters

Service for retrieving the list of filters that can be applied to Points Of Interest

9.1 Request

- (GET) /pois.filters.json

9.2 Parameters

- api_key: <string> API key (**required**)

9.3 Response

9.3.1 JSON

JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate.

The service will return in a JSON (JavaScript Object Notation) format.

```
{
  "filters": [{
    "values": [{
      "id": 1,
      "name": "Informal"
    },
    {
      "id": 2,
      "name": "Young"
    },
    {
      "id": 3,
      "name": "Familiar"
    },
    {
      "id": 4,
      "name": "Selected"
    },
    {
      "id": 5,
      "name": "Sophisticated"
    },
    {
      "id": 6,
      "name": "Historic"
    },
    {
      "id": 7,
      "name": "Formal"
    },
    {
      "id": 8,
      "name": "In"
    },
    {
      "id": 9,
      "name": "Romantic"
    },
    {
      "id": 10,
      "name": "Business"
    },
    {
      "id": 11,
      "name": "Cosy"
    }
  ]
}]
```


9.3 Response

```
    }],  
    "key": "ambiences"  
  },  
  {  
    "values": [{  
      "id": 1,  
      "name": "Rustic"  
    },  
    {  
      "id": 2,  
      "name": "Modern"  
    },  
    {  
      "id": 3,  
      "name": "Simple"  
    },  
    {  
      "id": 4,  
      "name": "Beach"  
    },  
    {  
      "id": 5,  
      "name": "Traditional"  
    },  
    {  
      "id": 6,  
      "name": "Maritime"  
    },  
    {  
      "id": 7,  
      "name": "Exotic"  
    },  
    {  
      "id": 8,  
      "name": "Zen"  
    },  
    {  
      "id": 9,  
      "name": "Classic"  
    },  
    {  
      "id": 10,  
      "name": "Luxurious"  
    },  
    {  
      "id": 11,  
      "name": "Typical"  
    },  
    {  
      "id": 12,  
      "name": "Revivalist"  
    }],  
    "key": "decorations"  
  }  
}
```

9.3 Response

}