Documentation of reseller-API

Wayke - V0.1 - 30/6 - 2017

Introduction

This documentation is provided to give a first glance at how Wayke API is planned to work. A first release candidate will be available from the middle of august of 2017, and a first release version for production use will be available from september 1st, 2017.

Access and limits

Wayke API will be available for retailers that have registered with Wayke for API access. Each retailer need to provide one or more IP addresses that will be given access to the Wayke API, and will in return be given an authentication key that is used in the communication with Wayke API.

Costs

-

Data formats

Wayke API exposes data serialized as JSON.

Other useful information

- To use a custom domain for ex. media files, it's recommended to create a CNAME pointer for "cdn.wayke.se".
- Some information exposed by Wayke API is data that Wayke doesn't own - but is provided by external sources, and that Wayke has a right to use for a limited time. This data can not be stored more than 90 days, and this restriction is inherited down to users of Wayke API.
- So called "rich text" follows the standard for Commonmark (http://www.commonmark.org). As of this writing, the only data provided in this format is the longer description of an advertisement.

API for vehicle/advertisement listings

Endpoint provides a listing of vehicles/advertisements available for the requesting retailer. The endpoints support pagination, with a page size of 25 documents.

Request

Method: `GET`
URI: https://api.wayke.se/vehicles
Parameters:

```
token (string)
      Required. The authentication key for the current retailer.
query (string)
      Optional. Used for free text searches. Default is null/empty string.
page (int)
      Optional. Which page to retrieve documents for. Defaults to 0.
sort (string)
      Optional. Which sort order to use. Available sort orders are “relevance”,
“published-asc”, “published-desc”, “price-asc”, “price-desc”. Defaults to
“relevance”.
```

Filter parameters:

```
manufacturer (string)
modelSeries (string)
gearboxType (manuell/automatisk/variomatisk)
fuelType (bensin/diesel/laddhybrid)

Example:
https://api.wayke.se/vehicles?token=*&query=barn&manufacturer=volvo&gearboxType=aut
omatisk&fuelType=laddhybrid
```

Response

Status code: `200 OK`

Body:

```
{
  “successful”: bool,
  “response”: {
    “documents”: [
      {
        “id”: GUID,
        “createdAt”: datetime,
        “updatedAt”: datetime,
        “publishedAt”: datetime,
        “title”: string,
        “featuredImage”: string,
        “milage”: int,
        “price”: float,
        “previousPrice”: float,
        “shortDescription”: string,
        “manufacturer”: string,
        “modelSeries”: string,
        “modelYear”: int,
        “retailers”: [
          string,
          ...
        ]
      }
    ],
    “documentCount”: int,
    “pageCount”: int
  },
  “message”: string
}
```

Possible returned error codes:

`400 Bad Request, 401 Unauthorized, 503 Service Unavailable`

API for a single vehicle/advertisement document

Endpoint provides all available data for a single vehicle/advertisement.

Request

Method: `GET`
URI: https://api.wayke.se/vehicles/{vehicle-id}
Parameters:

```
token (string)
      Required. The authentication key for the current retailer.
```

Response

Status code: `200 OK`

Body:

```
{
  “successful”: bool,
  “response”: {
    “id”: GUID,
    “createdAt”: datetime,
    “updatedAt”: datetime,
    “publishedAt”: datetime,
    “title”: string,
    “registrationNumber”: string,
    “vinNumber”: string,
    “featuredImage”: string,
    “milage”: int,
    “price”: float,
    “previousPrice”: float,
    “shortDescription”: string,
    “description”: string,
    “manufacturer”: string,
    “manufactureYear”: int,
    “modelSeries”: string,
    “modelName”: string,
    “modelYear”: int,
    “equipmentLevel”: string,
    “gearboxType”: string,
    “gearboxName”: string,
    “fuelType”: string,
    “enginePower”: int,
    “media”: [
      {
        “type”: string,
        “url”: string
      },
      ...
    ],
    “options”: [
      string,
      ...
    ],
    “properties”: {
      propertyName: string,
      ...
    },
    “retailers”: [
      string,
      ...
    ]
  },
  “message”: string
}
```

Possible returned error codes:

`401 Unauthorized, 404 Not Found, 503 Service Unavailable`