

Alletiders API

Sales channel

Version 1.21

Alletiders Api

Version: 1.21
Date: 2014.09.12
Main Author: dtr@combine.dk

Document History

Date	Version	Initials	Description
2013.05.14	0.1	kje	Document creation
2013.05.21	0.2	dtr	Resources documentation
2013.05.31	0.21	dtr	Updated order resource to include completion state and receipt
2013.10.16	0.22	dtr	Added paymentType to order and order id is now optional
2013.10.23	0.3	jdj	Added barcode-section
2013.10.23	0.4	dtr	Added sales agreements section
2013.12.20	0.41	jdj	Added new error messages in barcode-section
2014.02.21	0.5	dtr	Remove order receipt resource and updated order resource description
2014.03.04	0.51	dtr	Minor improvements to documentation
2014.03.10	0.52	dtr	Updated order resource, added ticket identity to orderline.
2014.03.13	0.6	dtr	Added theme resource
2014.03.14	0.7	dtr	Improvements
2014.03.19	0.8	dtr	Further improvements, layout change
2014.03.31	1.0	dtr, jri	Finalization of spelling + layout
2014.04.04	1.01	jdj, jri, dtr	Spelling corrections, code formatting
2014.04.30	1.02	dtr	Removed product profile types
2014.06.10	1.03	kdr	Added provider name to product search results.
2014.06.17	1.04	kdr	Added geolocation to product search results. Product search, provider list and provider detail descriptions corrected.
2014.06.24	1.05	dtr	Clarified order resource GET & PUT requests
2014.06.30	1.06	dtr	Updated order-lines structure to include ticket identities. Affects order and order-line resources.

2014.07.09	1.07	dtr	added start & end date to order line if its a booking product
2014.08.18	1.08	kdr	Added related product search for shopping cart products.
2014.08.27	1.09	dtr	Corrected orderline,order resources with right geolocation format
2014.08.28	1.1	dtr	Added language resource
2014.09.10	1.2	dtr	Changed currency response for xml format (made no sense)
2014.09.12	1.21	kdr	Corrected description of shopping cart related product request.
2014.11.07	1.3	sny	Changed input and response for product allotments

Table of Contents

- [Document History](#)
- [Table of Contents](#)
- [Document purpose](#)
- [System Overview](#)
- [API](#)
 - [RESTful service](#)
 - [Security, authorization and identity](#)
 - [HTTP session](#)
 - [Input and Output formats](#)
 - [Error handling](#)
 - [Global parameters](#)
- [Sales channel](#)
 - [Resource Overview](#)
 - [Order](#)
 - [Products](#)
 - [Provider](#)
 - [Trip Planner](#)
 - [Barcodes](#)
 - [Language, Currency, Sales Agreements, Batch Requests](#)
 - [Order](#)
 - [Create a New Order](#)
 - [Update Order Contents](#)
 - [Get Order Details](#)
 - [Get Orderline](#)
 - [Delete Orderline](#)
 - [Payment Transaction](#)
 - [Get Payment Url and Form Fields](#)
 - [Payment Callback](#)
 - [Products](#)
 - [Product Search](#)
 - [Popular Product List](#)
 - [Product Allotment List](#)
 - [City List](#)
 - [Theme List](#)
 - [Shopping Cart Related Product List](#)
 - [Provider](#)
 - [Provider List](#)
 - [Provider Details](#)
 - [Trip Planner](#)
 - [Trip Planner Concept List](#)

[Trip Planner Route List](#)

[Trip Planner Route Details](#)

[Barcodes](#)

[Barcode Details and Usage History](#)

[Ticket Identity Details](#)

[Verify Ticket](#)

[Language, Currency, Sales Agreements, Batch Requests](#)

[Language](#)

[Currency](#)

[Sales Agreement](#)

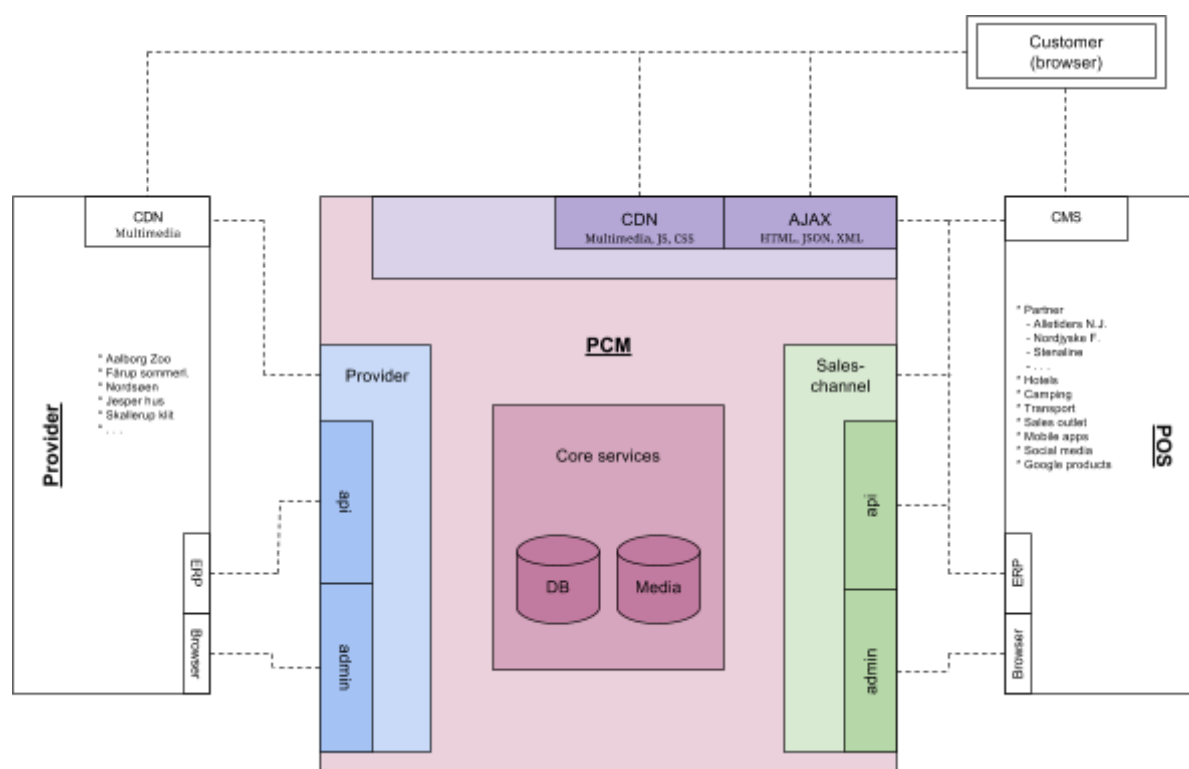
[Batch Request Grouping](#)

Document purpose

This document describes the API between sales channels and Alletiders API.

Please note: This is a living document, if the API changes it will updated accordingly and new versions issued.

System Overview



API

RESTful service

Alletiders API implements CRUD functionality on a series of resources allowing search of products, shopping cart and payment interactions. Enabling POS systems to display and make sales of products from various product providers.

Alletiders Api

Version: 1.21
 Date: 2014.09.12
 Main Author: dtr@combine.dk

Security, authorization and identity

The sales channel client is expected to use https and [basic authentication](#).

HTTP session

Order and payment interactions require session support from the channel client.

Input and Output formats

Data is available in JSON or XML formats.

All input parameters have to be passed as url encoded form - either in query string or POST/PUT data.

Presented examples contains additional white space for better readability, this is not the case for actual responses.

Error handling

If an exception occurs in any request, a special content is returned. This content will be either HTML, JSON, or XML and will depend on the current format. The content will include the actual error message and all exception information, including stack trace serialized to appropriate format. The stack trace and the error message as a whole is very useful for error investigation, please keep it, if you need to communicate with Combine about possible errors. The sales channel should be looking at the message(text in bold).

For example:

XML	<pre><?xml version="1.0" encoding="UTF-8"?> <error application="Product Channel Manager (PCM)"> <message>The requested page was not found</message> <exception> <message>Invalid controller specified (membuy)</message> <code>0</code> <stacktrace> ... </stacktrace> </exception> </error></pre>
JSON	<pre>{ "application": "Product Channel Manager (PCM)", "message": "The requested page was not found", "exception": { "message": "Invalid controller specified (membuy)", "code": 0, "stacktrace": [...] } }</pre>

	}
--	---

Global parameters

channel	API requires this parameter for most of the requests. It will be specified per request if it is required. The value is the name of the sales channel, case insensitive.
language	Language to use in ISO 639-1 standard, da, en (<i>optional</i>)
currency	The currency to use in ISO 4217 format. e.g. EUR, DKK. (<i>optional</i>) Must be one of the values returned by currency resource . If omitted default DKK is used.
format	Either JSON or XML. Behavior is undefined for other values. (<i>optional</i>) If omitted default JSON is used.

Sales channel

All prices in the responses are in cents, (currency * 100) to avoid rounding errors. Divide received numbers by 100 and format it before showing it to the user.

All input values that are to be used as currency have to be multiplied by 100.

Resource Overview

Order

Resource	Description
POST /channel/order	Create an order in the system. Results in a redirect to newly created resource.
PUT /channel/order	Update current order: change address, add products, reserve or complete an order. Results in the same response as in GET request, with updates performed.
GET /channel/order	Retrieve order resource, including order details, address, order lines, fees.

GET /channel/order-line	Retrieve order lines in current order
DELETE /channel/order-line/{id}	Delete order line by id from current order
POST /channel/payment-transaction	Create a new payment transaction
GET /channel/payment-transaction/{id}	Retrieve form fields required by payment provider
POST /channel/payment-callback	Invoked by payment-gateway

Products

Resource	Description
GET /channel/product	Retrieve a list of products matching search criterias
GET /channel/popular-product	Retrieve a list of most sold/popular products
GET /channel/product-allotment	Retrieve booking allotments
GET /channel/city	Retrieve a list of cities that has at least one product for sale currently
GET /channel/theme	Retrieve all active themes.
GET /channel/cart-related-products	Retrieve related products to products added to the shopping cart. The products are chosen by theme and city where theme has a priority. If there are no products from the same theme and city, then any other products will be returned. Products are returned randomly and there will not be the same products returned as in the shopping cart.

Provider

Resource	Description
GET /channel/provider	Retrieve a list of providers matching search criteria
GET /channel/provider/{id}	Retrieve provider details by id

Trip Planner

Resource	Description
----------	-------------

GET /channel/trip-planner-concept	Retrieve a list of concepts
GET /channel/trip-planner-route	Retrieve a list of routes
GET /channel/trip-planner-route/{id}	Retrieve route details by its id

Barcodes

Resource	Description
GET /channel/ticket-identity	Retrieve a list of ticket identities
GET /channel/ticket-identity/{id}	Retrieve ticket identity details including usage
POST /channel/ticket-identity/{id}	Scan or check single barcode

Language, Currency, Sales Agreements, Batch Requests

Resource	Description
GET /channel/language	Retrieve a list of supported languages
GET /channel/currency	Retrieve a list of supported currencies.
GET /channel/sales-agreement	Retrieve a list of active sales agreements.
GET /channel/batch	A special resource used to process multiple GET requests.

Order

Channel id is only required when you intend to create an order - that is only for POST request.

Create a New Order

POST /channel/order

Create a new order. Optionally you can specify parameters in the same structure as for update request to be used initially (You can save number of requests to the api).

Response code: **201 Created**

Location URI: /channel/order/{orderId}

Update Order Contents

PUT /channel/order

Updates an order, setting currency, address, orderlines etc. The order is identified via session cookie.

Parameters that you can specify:

currency	the currency to use in ISO 4217 format. e.g. EUR, DKK.
reserve	a boolean flag indicating that after this update order must be reserved, only applicable if order price = 0
complete	a boolean flag indicating that order was placed externally and it is not using the internal payment processing.
cancel	a boolean flag indicating that this order has to be cancelled. If 205 is returned, you can be sure that order will be cancelled. An error may occur only if ticket identities are received before this request is processed or if this order was not completed.
reason	optional message of why order is being cancelled, only used together with cancel parameter
address	a container element with: <ul style="list-style-type: none"> ● email* ● phone ● name* ● address* ● city* ● zipCode* ● country, default is Denmark Fields marked with * are required, if omitted, order update will fail.

products	<p>a collection of products that you wish to add to the order. Each product here is defined in this format:</p> <ul style="list-style-type: none"> • id - product identifier • qty - quantity, note that it cannot be changed later! • allotments - collection of key/value pairs. Pair key is allotment id and value has to be set to true. Allotments can be retrieved via ProductAllotment resource.
----------	--

Note that all orderlines are deleted when currency is changed!

Response code: **200 OK or 205 Reset Content**

Please see the next section “Get Order Details” for XML and JSON responses. Since the response for PUT and GET is the same. Although with one exception for 205 response code - in this case empty response is returned.

Get Order Details

GET /channel/order

Retrieves all the information about the order, including order details, currency, order lines, order fees and the address. The order itself is identified via session cookie.

Order lines, order fees, receipt and address may be empty.

Some explanations on fields:

receipt	<p>hash that can be used to show order receipt to the customer. It includes orderlines, tickets, and terms and conditions. If the order is not ready yet, it will show a “splash” page that will keep refreshing. It is meant to be accessed directly by the user. Url is as follows: order-receipt/{order.receipt} (no ‘channel’ in the beginning). The same url will be included in the email confirmation(if one is sent)</p>
paymentType	either full or fee
cost	cost of the order, based on product prices in sales agreements, always in DKK
isReady	a boolean flag indicating whether the order is completed and ticket identities were received/generated. It is safe to poll this resource.
isCancelled	a boolean flag indicating is the order was cancelled

Response code: **200 OK**

Example responses

XML	<pre><?xml version="1.0" encoding="UTF-8"?> <order xmlns="http://netimage.dk/pcm/channel/order" id="147"></pre>
-----	---

```

<insertTime>2013-05-16T10:21:47+02:00</insertTime>
<modified>2013-05-16T10:34:58+02:00</modified>
<price>6500</price>
<currency>DKK</currency>
<paymentType>full</paymentType>
<isCompleted>1</isCompleted>
<isCancelled>0</isCancelled>
<isReady>1</isReady>
<receipt></receipt>
<cost>1000</cost>
<address id="53">
  <name>dtr</name>
  <address>TestStreet 5</address>
  <zipCode>0000?</zipCode>
  <city>Test</city>
  <email>dtr@netimage.dk</email>
</address>
<orderLines>
  <orderLine id="764" productId="10">
    <title>1 dags leje af voksencykel (herre)</title>
    <quantity>1</quantity>
    <price>5000</price> <!-- quantity*unitPrice (including VAT) -->
    <cost>5000</cost>
    <unitPrice>5000</unitPrice>
    <coords>
      <latitude>57.024961</latitude>
      <longitude>9.940118</longitude>
    </coords>
    <ticketIdentities>
      <ticketIdentity id="39194">
        <type>barcode</type>
        <code>1564646579879</code>
        <maxUsage>1</maxUsage>
        <barcodeType>EAN13</barcodeType>
      </ticketIdentity>
      <ticketIdentity id="39195">
        <type>barcode</type>
        <code>4897984612311</code>
        <maxUsage>1</maxUsage>
        <barcodeType>EAN13</barcodeType>
      </ticketIdentity>
    </ticketIdentities>
    <!-- Only applicable to booking products -->
    <start>2014-07-19T00:00:00+02:00</start>
    <end>2014-07-20T00:00:00+02:00</end>
  </orderLine>
</orderLines>
<orderFees>
  <orderFee id="1" ver="1" recipientId="1">
    <msg>Bookinggebyr</msg>
    <value>1500</value>
  </orderFee>
</orderFees>

```

	<pre> <recipientName>VisitNordsjaelland</recipientName> </orderFee> </orderFees> </order> </pre>
JSON	<pre> { "id":147, "insertTime":"2013-05-16T10:21:47+02:00", "modified":"2013-05-16T10:34:58+02:00", "price":6500, "isCompleted":true, "isReady":true, "isCancelled":false, "receipt":null, "currency":"DKK", "paymentType":"full", "address":{ "id":53, "name":"dtr", "address":"TestStreet 5", "zipCode":"0000?", "city":"Test", "country":null, "phone":null, "email":"dtr@netimage.dk" }, "orderLines":[{ "id":764, "title":"1 dags leje af voksencykel (herre)", "quantity":1, "price":5000, // quantity*unitPrice (including VAT) "cost":1000, "unitPrice":5000, "productId":10, "orderLinePackageId":null, "orderLines":null, "coords":{"latitude":57.024961,"longitude":9.940118}, "ticketIdentities":[{ "id":39194, "type":"barcode", "code":"1564646579879", "message":null, "userId":null, "validDatePeriodId":null, "maxUsage":1, "barcodeType":"EAN13", "orderLineId":39218 }], { </pre>

	<pre> "id":39195, "type":"barcode", "code":"4897984612311", "message":null, "userId":null, "validDatePeriodId":null, "maxUsage":1, "barcodeType":"EAN13", "orderLineId":39218 }] }, "orderFees":[{ "id":1, "ver":1, "msg":"Bookinggebyr", "recipientId":1, "value":1500, "recipientName":"VisitNordsjaelland" }] } </pre>
--	---

Get Orderline

Channel id is not required for this resource.

GET /channel/order-line

Retrieves all the items in the order. The order is identified via session cookie. Data structure is the same as orderlines entry in the order.

Response code: **200 OK**

XML	<pre> <?xml version="1.0" encoding="UTF-8"?> <orderLines xmlns="http://netimage.dk/pcm/channel/order-line"> <orderLine id="764" productId="10"> <title>1 dags leje af voksencykel (herre)</title> <quantity>1</quantity> <price>5000</price> <cost>1000</cost> </orderLine> </orderLines> </pre>
-----	--

	<pre> <unitPrice>5000</unitPrice> <coords> <latitude>57.024961</latitude> <longitude>9.940118</longitude> </coords> <ticketIdentities> <ticketIdentity id="39194"> <type>barcode</type> <code>1564646579879</code> <maxUsage>1</maxUsage> <barcodeType>EAN13</barcodeType> </ticketIdentity> <ticketIdentity id="39195"> <type>barcode</type> <code>4897984612311</code> <maxUsage>1</maxUsage> <barcodeType>EAN13</barcodeType> </ticketIdentity> </ticketIdentities> </orderLine> </orderLines> </pre>
JSON	<pre> [{ "id":764, "title":"1 dags leje af voksencykel (herre)", "quantity":1, "price":5000, "cost":1000, "unitPrice":5000, "productId":10, "orderLinePackageId":null, "orderLines":null, "coords":{"latitude":57.024961,"longitude":9.940118}, "ticketIdentities":[{ "id":39194, "type":"barcode", "code":"1564646579879", "message":null, "userId":null, "validDatePeriodId":null, "maxUsage":1, "barcodeType":"EAN13", "orderLineId":39218 }, { "id":39195, "type":"barcode", "code":"4897984612311", "message":null, "userId":null, "validDatePeriodId":null, </pre>

	<pre> "maxUsage":1, "barcodeType":"EAN13", "orderLineId":39218 }] </pre>
--	---

Delete Orderlinde

DELETE /channel/order-line/{id}

Delete order line by id (parameter name:id) The order is identified via session cookie.

NOTE: Orderline id may contain multiple commas.

Response code: **200 OK**

Payment Transaction

POST /channel/payment-transaction

Create a new payment transaction. Call this when user intends to pay for the order. Only if payment will be handled by the API, otherwise use 'complete=1' to finalize the order.

Response code: **201 Created**

Location URI: /channel/payment-transaction/{transactionId}

Get Payment Url and Form Fields

GET /channel/payment-transaction/{transactionId}

Retrieve payment action url and required form fields.

3 parameters are accepted and required:

continueUrl	absolute url of where to redirect user after completing payment
cancelUrl	absolute url of where to redirect user after cancelling payment
ip	client ip address

The data received is only supposed to be used as is and result in a form that the user can submit. The action specified in response must be set as html form action, so the user would be directed to payment gateway. Any modification to form fields will result in an error in payment gateway.

Response code: **200 OK**

Example responses

XML	<pre><?xml version="1.0" encoding="UTF-8"?> <transaction xmlns="http://netimage.dk/pcm/channel/payment-transaction"> <action>https://secure.quickpay.dk/form/</action> <fields> <protocol>3</protocol> <msgtype>authorize</msgtype> <merchant>32499171</merchant> <language>da</language> <ordernumber>an_dtr_dev_test_72</ordernumber> <amount>6500</amount> <currency>DKK</currency> <continueurl>http://www.google.com</continueurl> <cancelurl>http://www.google.com</cancelurl> <callbackurl>http://dev.alletidersapi.com/channel/payment- callback?channel=Drupal</callbackurl> <autocapture>0</autocapture> <description></description> <ipaddress>192.168.0.1</ipaddress> <md5check>e72db3f0a80e28b4e2c6c7eb16b57679</md5check> </fields> </transaction></pre>
JSON	<pre>{ "action": "https:\\\\secure.quickpay.dk\\/form\\/", "fields": { "protocol": "3", "msgtype": "authorize", "merchant": "32499171", "language": "da", "ordernumber": "an_dtr_dev_test_72", "amount": 6500, "currency": "DKK", "continueurl": "http:\\\\www.google.com", "cancelurl": "http:\\\\www.google.com", "callbackurl": "http:\\\\dev.alletidersapi.com\\/channel\\/payment- callback?channel=Drupal", "autocapture": "0", "description": "", "ipaddress": "192.168.0.1", "md5check": "e72db3f0a80e28b4e2c6c7eb16b57679" } }</pre>

Payment Callback

POST /channel/payment-callback

Alletiders Api

Version: 1.21
 Date: 2014.09.12
 Main Author: dtr@combine.dk

This resource is only supposed to process data received from payment gateway. Parameters and outcome is implementation depended.

No channel id or authentication is required.

Response code: **200 OK**

Products

Product Search

GET /channel/product/

Retrieve a list of products matching search criteria.

Multiple optional parameters are available to filter the result:

types	An array or comma separated list: [standard booking package group]
city	City id, value can be retrieved via city resource
theme	Theme id, value can be retrieved via theme resource
provider	Provider id, value can be retrieved via provider resource
ids	An array or comma separated list of ids, only these specific products will be included in the result and if they match other rules.
offset	number of product to skip. Default is 0;
limit	number of products to find. Default is 10
startDate	starting date to filter allotments. In Y-m-d format. E.g. 2013-05-21
endDate	ending date to filter allotments. In Y-m-d format. E.g. 2014-01-21
modified	modified since date or array with start and end values. Format is mysql datetime, eg. 2013-02-13 09:54:26
random	boolean flag, if this is specified, resulting products will be sorted randomly. Default value is 0.
accommodations	Special value, Default is -1. if the value is 1, only booking products (that are accommodations) will be returned. If value is 0, booking products (but not accommodations) will be returned. Otherwise result is not affected.

includeAllotments	if this is specified, resulting products will have allotments inlined according to the startDate and endDate.
-------------------	---

3 properties may be empty: externalUrl, images, allotments

Response code: **200 OK**

Example responses

XML	<pre> <?xml version="1.0" encoding="UTF-8"?> <products xmlns="http://netimage.dk/pcm/channel/product" count="12"> <product id="9" ver="5" providerId="8"> <insertTime>2013-02-07T13:51:43+01:00</insertTime> <salesEmail>khe@netimage.dk</salesEmail> <coords> <latitude>57.024961</latitude> <longitude>9.940118</longitude> </coords> <title>Entre på Gilleleje Museum - 1 voksen</title> <intro>&lt;p&gt;&#xD; Kort beskrivelse af indgangsbilletten.&lt;/p&gt;</intro> <description>&lt;p&gt;&#xD; L&amp;aelig;ngere beskrivelse af indgangsbilletten.&lt;/p&gt;</description> <price>7500</price> <fullPrice>7500</fullPrice> <intervalUnit>day</intervalUnit> <intervalMin>1</intervalMin> <intervalMax>1</intervalMax> <type>booking</type> <providerName>Test udbyder</providerName> <images> <image> <thumb>http://dev.alletidersapi.com/upload/00/000/thumbs/03d.jpg</thumb> <main>http://dev.alletidersapi.com/upload/00/000/main/03d.jpg</main> <org>http://dev.alletidersapi.com/upload/00/000/03d.jpg</org> </image> </images> </product> <product id="10" ver="2" providerId="9"> <insertTime>2013-02-07T13:57:29+01:00</insertTime> <salesEmail>khe@netimage.dk</salesEmail> <title>1 dags leje af voksencykel (herre)</title> <intro>&lt;p&gt;&#xD; Beskrivelse af den fine herrecykel&lt;/p&gt;</intro> <description>&lt;p&gt;&#xD; L&amp;aelig;ngere beskrivelse af cyklen.&lt;/p&gt;</description> <price>5000</price> <fullPrice>5000</fullPrice> </pre>
-----	--

```

<intervalUnit>day</intervalUnit>
<intervalMin>1</intervalMin>
<intervalMax>1</intervalMax>
<type>booking</type>
<providerName>Test udbyder</providerName>
<images>
  <image>

<thumb>http://dev.alletidersapi.com/upload/00/000/thumbs/050.jpg</thumb>
  <main>http://dev.alletidersapi.com/upload/00/000/main/050.jpg</main>
  <org>http://dev.alletidersapi.com/upload/00/000/050.jpg</org>
</image>
</images>
<allotments>
  <allotment id="3633">
    <bookingEventId>183</bookingEventId>
    <start>2013-05-22T01:00:00+02:00</start>
    <end>2013-05-23T01:00:00+02:00</end>
    <availableTotal>1000</availableTotal>
    <isClosed>0</isClosed>
    <available>1000</available>
  </allotment>
</allotments>
</product>
<!-- Accommodation product with change-dates on Saturday (1 week) and monday
(4 days) -->
<product id="34" ver="3" providerId="6">
  <insertTime>2013-02-27T09:52:39+01:00</insertTime>
  <salesEmail>lan@visitnordsjaelland.com</salesEmail>
  <modifiedTime>
    <date>2013-10-16 13:27:17</date>
    <timezone_type>1</timezone_type>
    <timezone>+02:00</timezone>
  </modifiedTime>
  <title>Hytte</title>
  <intro>
    <p>Dejlig hytte til 4 personer</p>
  </intro>
  <description>
    <p>Dejlig hytte til 4 personer med udsigt over &oslash;resund</p>
  </description>
  <fullPrice>49800</fullPrice>
  <intervalUnit>day</intervalUnit>
  <intervalMin>1</intervalMin>
  <intervalMax>1</intervalMax>
  <type>booking</type>
  <providerName>Test udbyder</providerName>
  <allotments>
    <allotment id="5391,5392,5393,5394,5395,5396,5397">
      <bookingEventId>208</bookingEventId>
      <start>2013-10-19T00:00:00+02:00</start>
      <end>2013-10-26T00:00:00+02:00</end>
    </allotment>
  </allotments>
</product>

```

	<pre> <availableTotal>25</availableTotal> <isClosed>0</isClosed> <available>25</available> </allotment> <allotment id="5398,5399,5400,5401,5402,5403,5404"> <bookingEventId>208</bookingEventId> <start>2013-10-26T00:00:00+02:00</start> <end>2013-11-02T00:00:00+01:00</end> <availableTotal>25</availableTotal> <isClosed>0</isClosed> <available>25</available> </allotment> <allotment id="5393,5394,5395,5396"> <bookingEventId>208</bookingEventId> <start>2013-10-21T00:00:00+02:00</start> <end>2013-10-25T00:00:00+02:00</end> <availableTotal>25</availableTotal> <isClosed>0</isClosed> <available>25</available> </allotment> <allotment id="5400,5401,5402,5403"> <bookingEventId>208</bookingEventId> <start>2013-10-28T00:00:00+01:00</start> <end>2013-11-01T00:00:00+01:00</end> <availableTotal>25</availableTotal> <isClosed>0</isClosed> <available>25</available> </allotment> </allotments> <images> <image> <thumb>http://dev-cpm.dk/upload/00/000/thumbs/12a.png</thumb> <main>http://dev-cpm.dk/upload/00/000/main/12a.png</main> <org>http://dev-cpm.dk/upload/00/000/12a.png</org> </image> </images> </product> </products> </pre>
JSON	<pre> { "products": [{ "id": 9, "ver": 5, "providerId": 8, "insertTime": "2013-02-07T13:51:43+01:00", "salesEmail": "khe@netimage.dk", "coords": {"latitude": 57.024961, "longitude": 9.940118}, "externalUrl": null, "title": "Entre p\u00e5 Gilleleje Museum - 1 voksen", "intro": "<p>\r\n\tKort beskrivelse af indgangsbilletten.</p>\r\n", </pre>

```

        "description": "<p>\r\n\tL&aelig;ngere beskrivelse af
indgangsbilletten.</p>\r\n",
        "price": 7500,
        "fullPrice": 7500,
        "intervalUnit": "day",
        "intervalMin": 1,
        "intervalMax": 1,
        "type": "booking",
        "providerName": "Test udbyder",
        "images": [
            {
                "thumb": "http://dev.alletidersapi.com/upload/00/000/thumbs/03d.jpg",
                "main": "http://dev.alletidersapi.com/upload/00/000/main/03d.jpg",
                "org": "http://dev.alletidersapi.com/upload/00/000/03d.jpg"
            }
        ],
        "allotments": [
            {
                "id": 2644,
                "bookingEventId": 176,
                "start": "2013-05-22T01:00:00+02:00",
                "end": "2013-05-23T01:00:00+02:00",
                "availableTotal": "75",
                "isClosed": "0",
                "available": 75
            }
        ]
    },
    {
        "id": 10,
        "ver": 2,
        "providerId": 9,
        "insertTime": "2013-02-07T13:57:29+01:00",
        "salesEmail": "khe@netimage.dk",
        "externalUrl": null,
        "title": "1 dags leje af voksencykel (herre)",
        "intro": "<p>\r\n\tBeskrivelse af den fine herrecykel</p>\r\n",
        "description": "<p>\r\n\tL&aelig;ngere beskrivelse af cyklen.</p>\r\n",
        "price": 5000,
        "fullPrice": 5000,
        "intervalUnit": "day",
        "intervalMin": 1,
        "intervalMax": 1,
        "type": "booking",
        "providerName": "Test udbyder",
        "images": [
            {
                "thumb": "http://dev.alletidersapi.com/upload/00/000/thumbs/050.jpg",

```

```

"main": "http://dev.alletidersapi.com/upload/00/000/main/050.jpg",
"org": "http://dev.alletidersapi.com/upload/00/000/050.jpg"
    }
  ],
},
{
  "id": 34,
  "ver": 3,
  "providerId": 6,
  "label": null,
  "insertTime": "2013-02-27T09:52:39+01:00",
  "salesEmail": "lan@visitnordsjaelland.com",
  "modifiedTime": {
    "date": "2013-10-16 13:27:17",
    "timezone_type": 1,
    "timezone": "+02:00"
  },
  "coords": null,
  "externalUrl": null,
  "title": "Hytte",
  "intro": "<p>\r\n\tDejlig hytte til 4 personer</p>\r\n",
  "description": "<p>\r\n\tDejlig hytte til 4 personer med udsigt over
&Oslash;resund</p>\r\n",
  "price": null,
  "fullPrice": 49800,
  "intervalUnit": "day",
  "intervalMin": 1,
  "intervalMax": 1,
  "type": "booking",
  "providerName": "Test udbyder",
  "allotments": {
    "2013-10-19 00:00:00_ef3f331fa82fd2a7666f438a21c3307e": {
      "id": "5391,5392,5393,5394,5395,5396,5397",
      "bookingEventId": 208,
      "start": "2013-10-19T00:00:00+02:00",
      "end": "2013-10-26T00:00:00+02:00",
      "availableTotal": "25",
      "isClosed": "0",
      "available": 25
    },
    "2013-10-26 00:00:00_117c4c3d0b64dcb3a1a418c09b7be1e7": {
      "id": "5398,5399,5400,5401,5402,5403,5404",
      "bookingEventId": 208,
      "start": "2013-10-26T00:00:00+02:00",
      "end": "2013-11-02T00:00:00+01:00",
      "availableTotal": "25",
      "isClosed": "0",
      "available": 25
    },
    "2013-10-21 00:00:00_d6297c6db22b92372d99784e8f67430d": {
      "id": "5393,5394,5395,5396",

```


	<pre> "bookingEventId":208, "start":"2013-10-21T00:00:00+02:00", "end":"2013-10-25T00:00:00+02:00", "availableTotal":"25", "isClosed":"0", "available":25 }, "2013-10-28 00:00:00_5eeee4a0f922508a76c03d7188b56423":{ "id":"5400,5401,5402,5403", "bookingEventId":208, "start":"2013-10-28T00:00:00+01:00", "end":"2013-11-01T00:00:00+01:00", "availableTotal":"25", "isClosed":"0", "available":25 } }, "images":[{ "thumb":"http://dev-cpm.dk/upload/00/000/thumbs/12a.png", "main":"http://dev-cpm.dk/upload/00/000/main/12a.png", "org":"http://dev-cpm.dk/upload/00/000/12a.png" }] },], "productsCount":12 } </pre>
--	--

Popular Product List

GET /channel/popular-product

Retrieve a list of popular products.

4 optional parameters are available:

limit	Maximum number of popular products to find. Default is 5.
days	Number of days to look in the past. Default is 15.
city	City id to filter providers that product belongs to.
theme	Theme id to filter products.

Response code: **200 OK**

Example responses

Alletiders Api

Version: 1.21
 Date: 2014.09.12
 Main Author: dtr@combine.dk

XML	<pre><?xml version="1.0" encoding="UTF-8"?> <popularProducts xmlns="http://netimage.dk/pcm/channel/popular-product"> <popularProduct id="18" providerId="10"> <title>Entrebillet - Det Tekniske Museum - 1 Voksen</title> <lastSaleDate>2013-02-26T14:40:58+01:00</lastSaleDate> <numberOfSales>44</numberOfSales> </popularProduct> <popularProduct id="10" providerId="9"> <title>1 dags leje af voksencykel (herre)</title> <lastSaleDate>2013-02-26T14:40:58+01:00</lastSaleDate> <numberOfSales>30</numberOfSales> </popularProduct> <popularProduct id="11" providerId="9"> <title>1 dags leje af voksencykel (dame)</title> <lastSaleDate>2013-02-26T14:40:58+01:00</lastSaleDate> <numberOfSales>27</numberOfSales> </popularProduct> <popularProduct id="12" providerId="9"> <title>1 dags leje af barnencykel (pige)</title> <lastSaleDate>2013-02-26T14:40:58+01:00</lastSaleDate> <numberOfSales>24</numberOfSales> </popularProduct> <popularProduct id="25" providerId="17"> <title>Enkeltværelse</title> <lastSaleDate>2013-02-26T14:30:12+01:00</lastSaleDate> <numberOfSales>18</numberOfSales> </popularProduct> </popularProducts></pre>
JSON	<pre>[{ "id":18, "title":"Entrebillet - Det Tekniske Museum - 1 Voksen", "lastSaleDate":"2013-02-26T14:40:58+01:00", "numberOfSales":"44", "providerId":10 }, { "id":10, "title":"1 dags leje af voksencykel (herre)", "lastSaleDate":"2013-02-26T14:40:58+01:00", "numberOfSales":"30", "providerId":9 }, { "id":11, "title":"1 dags leje af voksencykel (dame)", "lastSaleDate":"2013-02-26T14:40:58+01:00", "numberOfSales":"27", "providerId":9 }, { "id":12, "title":"1 dags leje af barnencykel (pige)", "lastSaleDate":"2013-02-26T14:40:58+01:00", "numberOfSales":"24", "providerId":9 }, { "id":25, "title":"Enkeltværelse", "lastSaleDate":"2013-02-26T14:30:12+01:00", "numberOfSales":"18", "providerId":17 }]</pre>

	<pre> { "id":12, "title":"1 dags leje af barnecykel (pige)", "lastSaleDate":"2013-02-26T14:40:58+01:00", "numberOfSales":"24", "providerId":9 }, { "id":25, "title":"Enkeltev\u00e6relse", "lastSaleDate":"2013-02-26T14:30:12+01:00", "numberOfSales":"18", "providerId":17 }] </pre>
--	---

Product Allotment List

GET /channel/product-allotment

Retrieve a list of product allotments:

product	Product id. Required.
startDate	Starting date for allotments, default is current date.
limit	Number of days to get allotments for, will be added to startDate.
*Interval	Length of interval to fetch
*IntervalUnit	Unit of interval. Accepted values are: hour, day, week

*Accommodation products only

Response code: **200 OK**

Example responses

XML	<pre> <?xml version="1.0" encoding="UTF-8"?> <allotments xmlns="http://netimage.dk/pcm/channel/product-allotment" count="63"> <allotment id="3340"> <bookingEventId>181</bookingEventId> <start>2013-05-22T01:00:00+02:00</start> <end>2013-05-23T01:00:00+02:00</end> <available>15</available> </allotment> <allotment id="3341"> <bookingEventId>181</bookingEventId> <start>2013-05-23T01:00:00+02:00</start> </pre>
-----	--

	<pre> <end>2013-05-24T01:00:00+02:00</end> <available>15</available> </allotment> <allotment id="3342"> <bookingEventId>181</bookingEventId> <start>2013-05-24T01:00:00+02:00</start> <end>2013-05-25T01:00:00+02:00</end> <available>15</available> </allotment> <allotment id="3343"> <bookingEventId>181</bookingEventId> <start>2013-05-25T01:00:00+02:00</start> <end>2013-05-26T01:00:00+02:00</end> <available>15</available> </allotment> <allotment id="3344"> <bookingEventId>181</bookingEventId> <start>2013-05-26T01:00:00+02:00</start> <end>2013-05-27T01:00:00+02:00</end> <available>15</available> </allotment> <allotment id="3345"> <bookingEventId>181</bookingEventId> <start>2013-05-27T01:00:00+02:00</start> <end>2013-05-28T01:00:00+02:00</end> <available>15</available> </allotment> <allotment id="3346"> <bookingEventId>181</bookingEventId> <start>2013-05-28T01:00:00+02:00</start> <end>2013-05-29T01:00:00+02:00</end> <available>15</available> </allotment> </allotments> </pre>
JSON	<p>Accommodation allotments:</p> <pre> { "allotments": { "2014-11-10": { "id": "4790,4791,4792,4793", "bookingEventId": 135, "start": "2014-11-10T00:00:00+01:00", "end": "2014-11-14T00:00:00+01:00", "intervalUnit": "day", "availableTotal": "15", "isClosed": "0", </pre>

```

    "available":15,
    "combinationId":"4790,4791,4792,4793",
    "price":65000
  },
  "2014-11-24":
  {
    "id":"4889,4890,4891,4892",
    "bookingEventId":137,
    "start":"2014-11-24T00:00:00+01:00",
    "end":"2014-11-28T00:00:00+01:00",
    "intervalUnit":"day",
    "availableTotal":"10",
    "isClosed":"0",
    "available":10,
    "combinationId":"4889,4890,4891,4892",
    "price":65000
  },
  "2014-11-11":{
    "id":"4791,4792,4793,4794",
    "bookingEventId":135,
    "start":"2014-11-11T00:00:00+01:00",
    "end":"2014-11-15T00:00:00+01:00",
    "intervalUnit":"day",
    "availableTotal":"15",
    "isClosed":"0",
    "available":15,
    "combinationId":"4791,4792,4793,4794",
    "price":65000
  }
}, {"allotmentsCount":3
}

```

Booking allotments:

```

{
  "allotments":
  {
    "2014-11-10":
    {
      "2014-11-10 00:00:00":
      {
        "id":5012,
        "bookingEventId":158,
        "start":"2014-11-10T00:00:00+01:00",
        "end":"2014-11-17T00:00:00+01:00",

```

	<pre> "available":4 } }, "2014-11-17": { "2014-11-17 00:00:00": { "id":5013, "bookingEventId":158, "start":"2014-11-17T00:00:00+01:00", "end":"2014-11-24T00:00:00+01:00", "available":5 } }, "2014-11-24": { "2014-11-24 00:00:00": { "id":5014, "bookingEventId":158, "start":"2014-11-24T00:00:00+01:00", "end":"2014-12-01T00:00:00+01:00", "available":5 } }, "2014-12-01": { "2014-12-01 00:00:00":{ "id":5015, "bookingEventId":158, "start":"2014-12-01T00:00:00+01:00", "end":"2014-12-08T00:00:00+01:00", "available":5 } }, "2014-12-08": { "2014-12-08 00:00:00": { "id":5016, "bookingEventId":158, "start":"2014-12-08T00:00:00+01:00", "end":"2014-12-15T00:00:00+01:00", "available":5 } }, "2014-12-15": { "2014-12-15 00:00:00": </pre>
--	---

	<pre> { "id":5017, "bookingEventId":158, "start":"2014-12-15T00:00:00+01:00", "end":"2014-12-22T00:00:00+01:00", "available":5 } }, "allotmentsCount":6 } </pre>
--	--

City List

GET /channel/city

Retrieve a list of cities that have at least one product for sale at this point in time. Product can be filtered using values from this resource.

Response code: **200 OK**

Example responses

XML	<pre> <?xml version="1.0" encoding="UTF-8"?> <cities xmlns="http://netimage.dk/pcm/channel/city"> <city id="2439"> <title>Gilleleje</title> </city> <city id="2471"> <title>Helsinge</title> </city> <city id="2344"> <title>Helsingør</title> </city> <city id="2298"> <title>Hillerød</title> </city> <city id="2319"> <title>Slangerup</title> </city> </cities> </pre>
JSON	<pre> { "2439":{ "id":2439, "title":"Gilleleje" }, </pre>

	<pre> "2471":{ "id":2471, "title":"Helsinge" }, "2344":{ "id":2344, "title":"Helsing\u00f8r" }, "2298":{ "id":2298, "title":"Hiller\u00f8d" }, "2319":{ "id":2319, "title":"Slangerup" } } </pre>
--	---

Theme List

GET /channel/theme

Retrieve a list of active themes. Products can be filtered using values from this resource. Each product can belong to multiple themes.

Channel id is not required.

Response code: **200 OK**

Example responses

XML	<pre> <?xml version="1.0" encoding="UTF-8"?> <themes xmlns="http://netimage.dk/pcm/channel/theme"> <theme id="1"> <name>Theme 1</name> <descriptionShort></descriptionShort> <descriptionLong></descriptionLong> </theme> <theme id="2"> <name>Theme 2</name> <descriptionShort></descriptionShort> <descriptionLong></descriptionLong> </theme> </themes> </pre>
JSON	<pre> { "1":{ "id":1, "name":"Theme 1", "descriptionShort":"", "descriptionLong":"" }, </pre>

	<pre> "2":{ "id":2, "name":"Theme 2", "descriptionShort":"", "descriptionLong":"" } </pre>
--	--

Shopping Cart Related Product List

GET /channel/cart-related-products

Retrieve a list of products related to the products added to the shopping cart.

1 optional parameter is available:

limit	Maximum number of related products to return. Default is 4.
-------	---

Response code: **200 OK**

Example responses

XML	<pre> <?xml version="1.0" encoding="UTF-8"?> <cart-related-products xmlns="http://netimage.dk/pcm/channel/cart-related-products" count="4"> <product id="4" ver="3" providerId="1"> <insertTime>2013-07-09T12:58:01+02:00</insertTime> <salesEmail>khe@netimage.dk</salesEmail> <modifiedTime>2014-08-12T09:40:45+02:00</modifiedTime> <lockedInAdmin>0</lockedInAdmin> <isFree>0</isFree> <title>Kantur - test</title> <intro>test</intro> <description>&lt;p&gt;&#13; test&lt;/p&gt;&#13; </description> <providerName>Test produktudbyder</providerName> <type>theme</type> <isAccommodation>0</isAccommodation> <images> <image> <thumb>http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/thumbs/02d.jpg</thumb> <main>http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/main/02d.jpg</main> </pre>
-----	---

```

<org>http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/02d.jpg</org>
>
  </image>
</images>
</product>
<product id="6" ver="5" providerId="1">
  <insertTime>2013-07-09T15:43:10+02:00</insertTime>
  <salesEmail>kdr@netimage.dk</salesEmail>
  <modifiedTime>2014-06-18T15:57:09+02:00</modifiedTime>
  <lockedInAdmin>0</lockedInAdmin>
  <isFree>0</isFree>
  <title>Ørken</title>
  <intro>Ørken...</intro>
  <description>&lt;p&gt;&#13;
    Stor &oslash; rken..&lt;/p&gt;&#13;
  </description>
  <providerName>Test produktudbyder</providerName>
  <type>theme</type>
  <isAccommodation></isAccommodation>
  <images>
    <image>

<thumb>http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/thumbs/023
.jpg</thumb>

<main>http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/main/023.jp
g</main>

<org>http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/023.jpg</org>
>
  </image>
  <image>

<thumb>http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/thumbs/022
.jpg</thumb>

<main>http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/main/022.jp
g</main>

<org>http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/022.jpg</org>
>
  </image>
</images>
</product>
<product id="1" ver="1" providerId="1">
  <insertTime>2013-07-04T17:15:27+02:00</insertTime>
  <salesEmail>khe@netimage.dk</salesEmail>
  <modifiedTime>2014-07-22T13:41:55+02:00</modifiedTime>
  <lockedInAdmin>0</lockedInAdmin>
  <isFree>0</isFree>
  <title>Sommerhus</title>

```

	<pre> <intro>test</intro> <description>&lt;p>&#13; test&lt;/p>&#13; </description> <providerName>Test produktudbyder</providerName> <type>city</type> <isAccommodation>1</isAccommodation> <images> <image> <thumb>http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/thumbs/02c .jpg</thumb> <main>http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/main/02c.jp g</main> <org>http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/02c.jpg</org > </image> </images> </product> <product id="25" ver="2" providerId="1"> <insertTime>2013-09-26T12:37:34+02:00</insertTime> <salesEmail>khe@netimage.dk</salesEmail> <modifiedTime>2014-08-06T10:27:05+02:00</modifiedTime> <lockedInAdmin>0</lockedInAdmin> <isFree>0</isFree> <title>Bookingprodukt - test</title> <intro>Kort beskrivelse af produktet</intro> <description>&lt;p>&#13; test&lt;/p>&#13; </description> <providerName>Test produktudbyder</providerName> <type>other</type> <isAccommodation>0</isAccommodation> <images> <image> <thumb>http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/thumbs/02e .JPG</thumb> <main>http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/main/02e.JP G</main> <org>http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/02e.JPG</org > </image> </images> </product> </cart-related-products> </pre>
JSON	[

```

{
  "id":25,
  "ver":2,
  "providerId":1,
  "label":null,
  "insertTime":"2013-09-26T12:37:34+02:00",
  "salesEmail":"khe@netimage.dk",
  "modifiedTime":"2014-08-06T10:27:05+02:00",
  "lockedInAdmin":0,
  "isFree":0,
  "title":"Bookingprodukt - test",
  "intro":"Kort beskrivelse af produktet",
  "description":"<p>\r\n\ttest</p>\r\n",
  "providerName":"Test produktudbyder",
  "type":"theme",
  "isAccommodation":"0",
  "images":[
    {
      "thumb":"http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/thumbs/02e.JPG",
      "main":"http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/main/02e.JPG",
      "org":"http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/02e.JPG"
    }
  ]
},
{
  "id":1,
  "ver":1,
  "providerId":1,
  "label":null,
  "insertTime":"2013-07-04T17:15:27+02:00",
  "salesEmail":"khe@netimage.dk",
  "modifiedTime":"2014-07-22T13:41:55+02:00",
  "lockedInAdmin":0,
  "isFree":0,
  "title":"Sommerhus",
  "intro":"test",
  "description":"<p>\r\n\ttest</p>\r\n",
  "providerName":"Test produktudbyder",
  "type":"theme",
  "isAccommodation":"1",
  "images":[
    {
      "thumb":"http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/thumbs/02c.jpg",
      "main":"http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/main/02c.jpg",
      "org":"http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/02c.jpg"
    }
  ]
}

```

```

    ]
  },
  {
    "id":6,
    "ver":5,
    "providerId":1,
    "label":null,
    "insertTime":"2013-07-09T15:43:10+02:00",
    "salesEmail":"kdr@netimage.dk",
    "modifiedTime":"2014-06-18T15:57:09+02:00",
    "lockedInAdmin":0,"isFree":0,
    "title":"\u00d8rken",
    "intro":"\u00f8rken...",
    "description":"<p>\r\n\tStor &oslash;rken.</p>\r\n",
    "providerName":"Test produktudbyder",
    "type":"city",
    "isAccommodation":"",
    "images":[
      {
        "thumb":"http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/thumbs/023.jpg",
        "main":"http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/main/023.jpg",
        "org":"http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/023.jpg"
      },
      {
        "thumb":"http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/thumbs/022.jpg",
        "main":"http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/main/022.jpg",
        "org":"http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/022.jpg"
      }
    ]
  },
  {
    "id":7,
    "ver":2,
    "providerId":1,
    "label":null,
    "insertTime":"2013-07-09T16:04:28+02:00",
    "salesEmail":"kdr@netimage.dk",
    "modifiedTime":"2014-06-18T15:59:48+02:00",
    "lockedInAdmin":0,
    "isFree":0,
    "title":"Koala",
    "intro":"test",
    "description":"<p>\r\n\tttest</p>\r\n",
    "providerName":"Test produktudbyder",
    "type":"other",
    "isAccommodation":"",

```

	<pre> "images": [{ "thumb": "http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/ thumbs/021.jpg", "main": "http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/ main/021.jpg", "org": "http://projects.dev/aktivnordjylland/trunk/public/upload/00/000/0 21.jpg" }] }] </pre>
--	---

Provider

Provider List

GET /channel/provider

Retrieve a list of product providers. Optionally filtering by city and products they have.
2 parameters are accepted:

city	City id, value can be retrieved via city resource
having	An array or comma separated values specifying product types. Only providers having at least one product of each given type is returned. Available types: accommodation

Currently only accommodation type is supported: so all providers or those who provide accommodations can be returned.

Response code: **200 OK**

Example responses

XML	<pre> <?xml version="1.0" encoding="UTF-8"?> <providers xmlns="http://netimage.dk/pcm/channel/provider"> <provider id="22"> <name>7-Eleven</name> <description>Benzin station med kiosk salg, beliggende ved den store rundkørsel i Frederiksværk.&#13; &#13; </description> <contact></contact> <hours>1/1-2013 - 31/12-2013&#13; Døgnåben</hours> </pre>
-----	--

	<pre> <coords> <latitude>-0.9809254406677402</latitude> <longitude>3.8935085179169826e101</longitude> </coords> <images> <image> <thumb>http://dev.alletidersapi.com/upload/00/000/thumbs/10b.Shelle 7 - eleven</thumb> <main>http://dev.alletidersapi.com/upload/00/000/main/10b.Shelle 7 - eleven</main> <org>http://dev.alletidersapi.com/upload/00/000/10b.Shelle 7 - eleven</org> </image> </images> </provider> <provider id="53"> <name>Adamsens Fisk</name> <description></description> <contact></contact> <hours></hours> <images> <image> <thumb>http://dev.alletidersapi.com/upload/00/000/thumbs/1fc.Adamsens Fisk</thumb> <main>http://dev.alletidersapi.com/upload/00/000/main/1fc.Adamsens Fisk</main> <org>http://dev.alletidersapi.com/upload/00/000/1fc.Adamsens Fisk</org> </image> <image> <thumb>http://dev.alletidersapi.com/upload/00/000/thumbs/1fb.Kærlighed på cykel i Kongernes Nordsjælland</thumb> <main>http://dev.alletidersapi.com/upload/00/000/main/1fb.Kærlighed på cykel i Kongernes Nordsjælland</main> <org>http://dev.alletidersapi.com/upload/00/000/1fb.Kærlighed på cykel i Kongernes Nordsjælland</org> </image> <image> <thumb>http://dev.alletidersapi.com/upload/00/000/thumbs/1fd.Langs den danske riviera</thumb> <main>http://dev.alletidersapi.com/upload/00/000/main/1fd.Langs den danske riviera</main> <org>http://dev.alletidersapi.com/upload/00/000/1fd.Langs den danske riviera</org> </image> </images> </provider> </providers> </pre>
JSON	<pre> [{ "id":22, "name":"7-Eleven", </pre>

```

        "description": "Benzin station med kiosk salg, beliggende ved den store
rundk\u00f8rsel i Frederiksv\u00e6rk.\r\n\r\n",
        "contact": "",
        "hours": "1\1-2013 - 31\12-2013\r\nD\u00f8gn\u00e5ben",
        "coords": {
            "latitude": "-0.9809254406677402",
            "longitude": "3.8935085179169826e101"
        },
        "images": [
            {
                "thumb": "http://dev.alletidersapi.com/upload/00/000/thumbs/10b.Shelle 7 -
eleven",
                "main": "http://dev.alletidersapi.com/upload/00/000/main/10b.Shelle 7 -
eleven",
                "org": "http://dev.alletidersapi.com/upload/00/000/10b.Shelle 7
-eleven"
            }
        ],
        {
            "id": 53,
            "name": "Adamsens Fisk",
            "description": "",
            "contact": "",
            "hours": "",
            "coords": null,
            "images": [
                {
                    "thumb": "http://dev.alletidersapi.com/upload/00/000/thumbs/1fc.Adamsens
Fisk",
                    "main": "http://dev.alletidersapi.com/upload/00/000/main/1fc.Adamsens
Fisk",
                    "org": "http://dev.alletidersapi.com/upload/00/000/1fc.Adamsens
Fisk"
                },
                {
                    "thumb": "http://dev.alletidersapi.com/upload/00/000/thumbs/1fb.K\u00e6rlig
hed p\u00e5 cykel i Kongernes Nordsj\u00e6lland",
                    "main": "http://dev.alletidersapi.com/upload/00/000/main/1fb.K\u00e6rlighed
p\u00e5 cykel i Kongernes Nordsj\u00e6lland",
                    "org": "http://dev.alletidersapi.com/upload/00/000/1fb.K\u00e6rlighed
p\u00e5 cykel i Kongernes Nordsj\u00e6lland"
                },
            ]
        }
    ]
}

```


	<pre> "thumb": "http://dev.alletidersapi.com/upload/00/000/thumbs/1fd.Langs den danske riviera", "main": "http://dev.alletidersapi.com/upload/00/000/main/1fd.Langs den danske riviera", "org": "http://dev.alletidersapi.com/upload/00/000/1fd.Langs den danske riviera" }] }] </pre>
--	--

Provider Details

GET /channel/provider/{providerId}

Retrieve provider details.

2 parameters are accepted:

Example responses

id	provider Id
type	if this is set to 'product', id is treated as product id. In other words, provider is found by product.

Response code: **200 OK**

XML	<pre> <?xml version="1.0" encoding="UTF-8"?> <provider xmlns="http://netimage.dk/pcm/channel/provider" id="6"> <name>Skjalm Hvide Hotel</name> <description>Et godt gammelt hotel.</description> <contact>Telefon: 56 32 65 23</contact> <hours>Åbent alle dage - ring for mere information</hours> <coords> <latitude>-0.00000000029045075507383443</latitude> <longitude>-1.8041802729387393e230</longitude> </coords> <images> <image> <thumb>http://dev.alletidersapi.com/upload/00/000/thumbs/006.png</thumb> <main>http://dev.alletidersapi.com/upload/00/000/main/006.png</main> <org>http://dev.alletidersapi.com/upload/00/000/006.png</org> </image> </images> </provider> </pre>
JSON	{

	<pre> {id":6, "name":"Skjalm Hvide Hotel", "description":"Et godt gammelt hotel.", "contact":"Telefon: 56 32 65 23", "hours":"\u00c5bent alle dage - ring for mere information", "coords":{ "latitude":"-0.00000000029045075507383443", "longitude":"-1.8041802729387393e230" }, "images":[{ "thumb":"http://dev.alletidersapi.com/upload/00/000/thumbs/006.png", "main":"http://dev.alletidersapi.com/upload/00/000/main/006.png", "org":"http://dev.alletidersapi.com/upload/00/000/006.png" }] } </pre>
--	--

Trip Planner

Trip Planner Concept List

GET /channel/trip-planner-concept

Retrieve a list of concepts (route groups) used by trip planner.

Response code: **200 OK**

Example responses

XML	<pre> <?xml version="1.0" encoding="UTF-8"?> <concepts xmlns="http://netimage.dk/pcm/channel/trip-planner-concept"> <concept id="3"> <isActive>1</isActive> <title>King's Challenge</title> </concept> <concept id="4"> <isActive>1</isActive> <title>Kærlighed på cykel</title> </concept> <concept id="5"> <isActive>1</isActive> <title>Dessertweekend</title> </concept> <concept id="6"> <isActive>1</isActive> <title>Langs den danske riviera</title> </concept> </concepts> </pre>
-----	--

	<code></concepts></code>
JSON	<pre>[{ "id":3, "isActive":1, "title":"King's Challenge" }, { "id":4, "isActive":1, "title":"K\u00e6rlighed p\u00e5 cykel" }, { "id":5, "isActive":1, "title":"Dessertweekend" }, { "id":6, "isActive":1, "title":"Langs den danske riviera" }]</pre>

Trip Planner Route List

GET /channel/trip-planner-route

Retrieve a list of routes used by trip planner. 1 parameter is accepted:

concept	Concept id. If this is left out, first concept is located and its routes are returned.
---------	--

Response code: **200 OK**

Example responses

XML	<pre><?xml version="1.0" encoding="UTF-8"?> <routes xmlns="http://netimage.dk/pcm/channel/trip-planner-route"> <route id="3"> <conceptId>3</conceptId> <isActive>1</isActive> <title>Family Explorer</title> </route> </routes></pre>
JSON	<pre>[{</pre>

	<pre> "id":3, "conceptId":3, "isActive":1, "fileId":null, "title":"Family Explorer", "description":null }] </pre>
--	--

Trip Planner Route Details

GET /channel/trip-planner-route/{routeId}

Retrieve route details.

Response code: **200 OK**

Example responses

XML	<pre> <?xml version="1.0" encoding="UTF-8"?> <route xmlns="http://netimage.dk/pcm/channel/trip-planner-route" id="3" conceptId="3"> <title>Family Explorer</title> <estimatedPrice>199500</estimatedPrice> <days> <day id="2" cityId="2439"> <products> <product>5</product> <product>9</product> <product>10</product> <product>11</product> <product>13</product> <product>18</product> <product>19</product> </products> </day> <day id="3" cityId="2448"> <products> <product>7</product> <product>8</product> </products> </day> <day id="4" cityId="2344"> <products> <product>11</product> <product>12</product> <product>13</product> <product>18</product> </products> </day> </days> </pre>
-----	---

	<code></route></code>
JSON	<pre>{ "id":3, "conceptId":3, "title":"Family Explorer", "description":null, "estimatedPrice":199500, "days":[{ "id":2, "cityId":2439, "products":[5, 9, 10, 11, 13, 18, 19] }, { "id":3, "cityId":2448, "products":[7, 8] }, { "id":4, "cityId":2344, "products":[11, 12, 13, 18] }] }</pre>

Barcodes

Barcode Details and Usage History

GET /channel/ticket-identity

Returns barcode history and details

Alletiders Api

Version: 1.21
 Date: 2014.09.12
 Main Author: dtr@combine.dk

Response code: **200 OK**

Example responses

XML	<pre><?xml version="1.0" encoding="UTF-8"?> <ticketIdentities xmlns="http://netimage.dk/pcm/channel/ticket-identity"> <ticketIdentity id="23"> <type>barcode</type> <code>8705647250635</code> <product id="6" ver="1"> <title>Dobbeltværelse (pris for 2 pers.)</title> <ticketDescription>Tillykke med dit køb. </ticketDescription> </product> <usage/> </ticketIdentity> <ticketIdentity id="805"> <type>barcode</type> <code>1234561234567</code> <validDatePeriodId>23</validDatePeriodId> <product id="35" ver="1"> <title>Entre</title> </product> <usage> <scan id="1"> <insertTime>2013-10-21T15:02:49+02:00</insertTime> <scannedBy>John Doe &lt;jd@example.com&gt;</scannedBy> </scan> <scan id="11"> <insertTime>2013-10-21T15:40:12+02:00</insertTime> <scannedBy>John Doe &lt;jd@example.com&gt;</scannedBy> </scan> </usage> </ticketIdentity> </ticketIdentities></pre>
JSON	<pre>[{ "id":25, "type":"barcode", "code":"8425201249386", "message":null, "userId":null, "validDatePeriodId":null, "maxUsage":1, "url":null, "product":{ "id":25, "ver":1, "title":"Bookingprodukt - test", "ticketDescription":"Test af billetbeskrivelsen, som skal st\u00e5 her." }, "usage":[</pre>

	<pre>], "validFrom":null, "validTo":null }] </pre>
--	--

Ticket Identity Details

GET /channel/ticket-identity/{id}

Retrieve ticket identity details.

1 parameter is accepted:

id	ticket barcode
----	----------------

Response code: **200 OK**

Example responses

XML	<pre> <?xml version="1.0" encoding="UTF-8"?> <ticketIdentity xmlns="http://netimage.dk/pcm/channel/ticket-identity"> <id>805</id> <type>barcode</type> <code>1234561234567</code> <validDatePeriodId>23</validDatePeriodId> <product id="35" ver="1"> <title>Entre</title> </product> <usage> <scan id="1"> <insertTime>2013-10-21T15:02:49+02:00</insertTime> <scannedBy>John Doe &lt;jd@example.com&gt;</scannedBy> </scan> <scan id="11"> <insertTime>2013-10-21T15:40:12+02:00</insertTime> <scannedBy>John Doe &lt;jd@example.com&gt;</scannedBy> </scan> </usage> </ticketIdentity> </pre>
JSON	<pre> { "id":25, "type":"barcode", "code":"8425201249386", "message":null, "userId":null, "validDatePeriodId":null, </pre>

	<pre> "maxUsage":1, "url":null, "product":{ "id":25, "ver":1, "title":"Bookingprodukt - test", "ticketDescription":"Test af billetbeskrivelsen, som skal st\u00e5 her." }, "usage":{ "3":{ "id":3, "insertTime":"2014-03-17T09:38:51+01:00", "revoke":null, "scannedBy":"Domas Trijonis <dtr@netimage.dk>" } }, "validFrom":null, "validTo":null } </pre>
--	---

Verify Ticket

POST /channel/ticket-identity/{id}

Tries to scan a given ticket. A list of errors will be returned. For more details overview, see the ticket identity read-action.

Channel id is not required for this resource.

2 parameters are accepted:

id	the barcode to scan
type	Set value to "scan" to mark the ticket as used, and "check" to only validate the barcode. Default is to scan the barcode.
provider	Provider id where ticket is scanned or checked. If omitted, it is your provider.

Response code: **200 Ok**

Example responses

XML	<pre> <?xml version="1.0" encoding="UTF-8"?> <errors xmlns="http://netimage.dk/pcm/channel/ticket-identity"/> </pre>
JSON	[0]

Response code: **420 unused**

Example responses

XML	<pre><?xml version="1.0" encoding="UTF-8"?> <error application="Product Channel Manager (PCM)"> <message>Data error</message> <exception> <message>The ticket has already been used at 2013-10-21 15:02:49 by Jean Connery from VisitNordsjælland</message> <code>3000</code> <details> <insertTime> <date>2013-10-21 15:02:49</date> <timezone_type>1</timezone_type> <timezone>+02:00</timezone> </insertTime> <user>Jan Rise</user> <organization>VisitNordsjælland</organization> </details> </exception> </error></pre>
JSON	<pre>{ "application": "Product Channel Manager (PCM)", "message": "rest.error.input.error", "exception": { "message": "The package the ticket belongs to is expired", "code": 2000 } }</pre>

Possible error codes:

2000 - The package the ticket belongs to is expired since the first use (for package products only)

2001 - The ticket itself has expired

3000 - The ticket has already been used the allowed number of times. Call read to get the complete list of usages.

3001 - The ticket has already been used the allowed number of times for the specific provider.

4000 - The ticket exists, but belongs to another organization

4500 - The ticket exists, but can only be used by different provider

5000 - The ticket does not exist

Language, Currency, Sales Agreements, Batch Requests

Language

GET /channel/language

Retrieve a list of supported languages.

Channel id is not required.

Response code: **200 OK**

Example responses

XML	<pre><?xml version="1.0" encoding="UTF-8"?> <languages xmlns="http://netimage.dk/pcm/channel/language"> <language id="1" code="da" locale="da_DK">danish</language> <language id="2" code="en" locale="en_GB">english</language> <language id="3" code="de" locale="de_DE">german</language> <language id="4" code="no" locale="nb_NO">norwegian</language> <language id="5" code="sv" locale="sw_SE">swedish</language> <language id="6" code="nl" locale="nl_NL">dutch</language> </languages></pre>
JSON	<pre>[{ "id":1, "label":"danish", "locale":"da_DK", "code":"da" }, { "id":2, "label":"english", "locale":"en_GB", "code":"en" }, { "id":3, "label":"german", "locale":"de_DE", "code":"de" }, { "id":4, "label":"norwegian", "locale":"nb_NO", "code":"no" }, { "id":5, "label":"swedish", "locale":"sw_SE", "code":"sv" }, { "id":6,</pre>

	<pre> "label": "dutch", "locale": "nl_NL", "code": "nl" }] </pre>
--	--

Currency

GET /channel/currency

Retrieve a list of supported currencies.

Channel id is not required.

Response code: **200 OK**

Example responses

XML	<pre> <?xml version="1.0" encoding="UTF-8"?> <currencies xmlns="http://netimage.dk/pcm/channel/currency"> <currency id="1">DKK</currency> <currency id="2">EUR</currency> <currency id="3">USD</currency> <currency id="4">SEK</currency> <currency id="5">NOK</currency> </currencies> </pre>
JSON	<pre> [{ "id": 1, "currency": "DKK" }, { "id": 2, "currency": "EUR" }, { "id": 3, "currency": "USD" }, { "id": 4, "currency": "SEK" }, { "id": 5, "currency": "NOK" }] </pre>

Sales Agreement

GET /channel/sales-agreement

Retrieve a list of active sales agreements.
Note that agreement may have no products.

Response code: **200 OK**

Example responses

XML	<pre><?xml version="1.0" encoding="UTF-8"?> <salesAgreements xmlns="http://netimage.dk/pcm/channel/salesAgreement"> <salesAgreement id="1"> <name>Unnamed agreement</name> <from>2013-10-22T00:00:00+02:00</from> <to>2013-11-22T00:00:00+01:00</to> <products> <product id="10" providerId="9" quantity="10" price="3500"> <title>1 dags leje af voksencykel (herre)</title> <providerName>Cykeludlejning - Danhostel Helsingør</providerName> </product> </products> </salesAgreement> <salesAgreement id="2"> <name>Unnamed agreement</name> <from>2013-10-23T00:00:00+02:00</from> <to>2014-05-02T00:00:00+02:00</to> <products> <product id="29" providerId="6" quantity="50" fullPrice="50000" price="20000"> <title>Morgenmad for 2 personer</title> <providerName>Skjalm Hvide Hotel</providerName> </product> </products> </salesAgreement> </salesAgreements></pre>
JSON	<pre>[{ "id": 1, "name": "Unnamed agreement", "from": "2013-10-22T00:00:00+02:00", "to": "2013-11-22T00:00:00+01:00", "products": [{ "id": 10, "title": "1 dags leje af voksencykel (herre)", "providerId": 9, "providerName": "Cykeludlejning - Danhostel Helsingør", "fullPrice": null,</pre>

	<pre> "price": 3500, "quantity": 10 }], }, { "id": 2, "name": "Unnamed agreement", "from": "2013-10-23T00:00:00+02:00", "to": "2014-05-02T00:00:00+02:00", "products": [{ "id": 29, "title": "Morgenmad for 2 personer", "providerId": 6, "providerName": "Skjalm Hvide Hotel", "fullPrice": 50000, "price": 20000, "quantity": 50 }] }] </pre>
--	---

Batch Request Grouping

GET /channel/batch

This is a special resource, all previously mentioned resources can be accessed through batch request. Purpose of this is to group requests into one request instead of making multiple http requests.

1 parameter is required:

r	<p>This is a key/value pairs of requests to be performed. Pair key is request id, and value is url encoded path.</p> <p>The path begins with resource name, not with 'channel/'.</p> <p>Global parameters (language, currency, channel) are set only to batch request (no need to duplicate them for each sub request).</p> <p>Only GET requests are supported.</p>
---	--

Response code: **200 OK**

Result of this is highly depended on the request. But in all cases it is a collections of requests, each having an id (pair key) and value being the result of the request.

XML	<pre><?xml version="1.0" encoding="UTF-8"?> <requests xmlns="http://netimage.dk/pcm/channel/batch"> <request id="cities">...</request> <request id="products">...</request> </requests></pre>
JSON	<pre>{ "cities":[....], "products":[...] }</pre>