

HKS External Service

Preface

The HKS External Service Interface (ExSI) is a protocol for REST web services external systems can use to access the HKS system. The ExSI protocol is implemented in the HKS system itself and may also be implemented in external systems. ExSI can further be implemented in message routers to enable more than two parties to communicate with each other.

Available Services

*Partner-Service: Service to get information on ExSI participants (partners), especially for message router scenarios.

*Catalog-Service: Service to get information about products that can be sold.

*Couponing-Service: Service to create and redeem coupons.

*Mediumdata-Service: Service to get information for media (barcode tickets, transponder chips, etc.) and its performances (access permissions, value cards, ...).

*POS-Service: Service for point-of-sale operations, e.g. product sale and cancel.

*Customer-Service: Service for customer operation, e.g. getting customer information and new customer creation.

*Version-Service: Get the web server's version and meta information.

*Script-Service: Commands for additional scripting.

*EMobility-Service: Commands for EMobility services.

Implementation

ExSI can be used via HTTP or HTTPS and on any port depending on the service's configuration. Message payloads are JSON-encoded. Date values are ISO 8601 or Unix timestamp encoded depending on configuration.

Authentication

ExSI uses HTTP Basic Auth. Use of HTTPS is recommended.

Message routing

ExSI messages may be routed by an ExSI compatible message router. This enables more than two systems to communicate with each other with only one connection (the one to the message router). The "Communication Center" is a message router. Every ExSI participant has a unique id (an integer value) which is set by configuration. ExSI messages can contain sender and addressee ids as HTTP header information. The sender id is sent as "X-ExSI-Server: <id>", the addressee's id as "X-ExSI-Addressee: <id>". When a participant sends a message to the message router, the sender id is

optional. In the message the router forwards to a receiver the sender id is mandatory. If the sender id does not match the authenticated user, the request is answered with error message <tbdl!>. The addressee's id is optional unless otherwise stated in the command's documentation. If the message router cannot determine a receiver, the request is answered with error message <tbdl!>.

Version History

Document Version	Author	Changes	Date
1.0	Michael Hirsch	Initial Version	2018-01-26
1.1	Michael Hirsch	Added Script-Service	2018-04-13
1.2	Michael Hirsch	Added Couponing-Service	2018-05-28
1.3	Michael Hirsch	Added EMobility-Service, Added inhouse credit query to Medium-Service	2018-10-15
1.4	Michael Hirsch	Added locker information to medium request	2019-01-11
1.5	Michael Hirsch	Improved sale endpoints	2019-03-26

Overview

Version information

Version : 1.5

URI scheme

Host : localhost:31100

BasePath : /

Schemes : HTTP

Tags

- catalog
- coupon
- customers
- emobility
- mediumdata
- partner

- pos
- script
- version

Consumes

- `application/json`

Produces

- `application/json`

Paths

Catalog: Request catalog information

GET /catalog

Description

Get catalog information, e.g. available products, tax information, etc.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Query	DeviceID <i>optional</i>	Optional parameter specifying the device the catalog is meant for	string

Responses

HTTP Code	Description	Schema
200	Success	CatalogResponse

Consumes

- `application/json`

Produces

- `application/json`

Tags

- catalog

Security

Type	Name
basic	basic

Coupons: Get a cart's rebated prices

POST /coupon/cart

Description

Post a coupon and a cart and get the cart's rebated prices.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Body	body <i>optional</i>		Cart

Responses

HTTP Code	Description	Schema
200	Success	CartResponse

Consumes

- `application/json`

Produces

- `application/json`

Tags

- coupon

Security

Type	Name
basic	basic

Coupons: Gets customer account coupons

```
GET /coupon/customer/{CustomerId}
```

Description

Gets info about coupons stored in a customer's account.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Path	CustomerId <i>required</i>	Coupon code	string

Responses

HTTP Code	Description	Schema
200	Success	CustomerCouponResponse

Consumes

- `application/json`

Produces

- `application/json`

Tags

- coupon

Security

Type	Name
basic	basic

Couponing: Create and return coupons

POST /coupon/order

Description

Returns coupons in regard to a customer's order and its grand total. Attention: this call is NOT idempotent! Store the results of the first call and don't call it more than once per order!

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Body	body <i>optional</i>		OrderCoupons

Responses

HTTP Code	Description	Schema
200	Success	OrderCouponsResponse

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- coupon

Security

Type	Name
basic	basic

Coupons: Redeems a coupon

POST /coupon/redeem/{Code}

Description

Redeems a coupon.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Path	Code <i>required</i>	Coupon code.	string
Body	body <i>optional</i>		RedeemCoupon

Responses

HTTP Code	Description	Schema
200	Success	RedeemCouponResponse

Consumes

- [application/json](#)

Produces

- `application/json`

Tags

- coupon

Security

Type	Name
basic	basic

Coupons: Gets a coupon's definition

GET /coupon/{Code}

Description

Gets a coupon's definition by its code.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Path	Code <i>required</i>	Coupon code	string

Responses

HTTP Code	Description	Schema
200	Success	CouponResponse

Consumes

- `application/json`

Produces

- `application/json`

Tags

- coupon

Security

Type	Name
basic	basic

Customers: Customer Search by properties

GET /customers

Description

Attention: DateFormat: 'mm.dd.yyyy HH:mm:ss' or 'mm-dd-yyyy HH:mm:ss'. E.g.: <http://IP/customers/06.27.2016%2011:12:00>.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Query	Birthdate <i>optional</i>		string (date-time)
Query	City <i>optional</i>	City	string
Query	FilterMainPerson <i>optional</i>	When set to true, returns customers with TheBear as the sole person. When set to false, a customer with all persons will be returned.	boolean
Query	Firstname <i>optional</i>		string
Query	Lastname <i>optional</i>	City	string
Query	Mediumnumber <i>optional</i>		integer (int32)

Type	Name	Description	Schema
Query	Number <i>optional</i>		string
Query	ResultLimit <i>optional</i>	Limits the number of returned results.	integer (int16)

Responses

HTTP Code	Description	Schema
200	Success	CustomersResponse

Consumes

- `application/json`

Produces

- `application/json`

Tags

- customers

Security

Type	Name
basic	basic

Customers: Flat Customer Creation

POST /customers/flat

Description

Creates a new customer.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Body	body <i>optional</i>		CreateFlatCustomer

Responses

HTTP Code	Description	Schema
200	Success	FlatCustomerResponse

Consumes

- `application/json`

Produces

- `application/json`

Tags

- customers

Security

Type	Name
basic	basic

Customers: Flat Customer Search by properties

GET /customers/flat

Description

Attention: DateFormat: 'mm.dd.yyyy HH:mm:ss' or 'mm-dd-yyyy HH:mm:ss'. E.g.: <http://IP/customers/flat?Number=1.10.3344>.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Query	Birthdate <i>optional</i>		string (date-time)
Query	City <i>optional</i>		string
Query	FilterMainPerson <i>optional</i>	When set to true, returns flat customers with TheBear as the sole person. When set to false, a flat customer for every person will be returned.	boolean
Query	Firstname <i>optional</i>		string
Query	Lastname <i>optional</i>		string
Query	Mediumnumber <i>optional</i>		integer (int32)
Query	Number <i>optional</i>		string
Query	ResultLimit <i>optional</i>	Limits the number of returned results.	integer (int16)

Responses

HTTP Code	Description	Schema
200	Success	FlatCustomersResponse

Consumes

- `application/json`

Produces

- `application/json`

Tags

- customers

Security

Type	Name
basic	basic

Customers: Flat Customer update

PUT /customers/flat

Description

Updates the customer. Attention: EMail will not be updated! If PersonId is NULL, a new Person will be created for this customer.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Body	body <i>optional</i>		UpdateFlatCustomer

Responses

HTTP Code	Description	Schema
200	Success	FlatCustomerResponse

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- customers

Security

Type	Name
basic	basic

Customers: Flat Customer Search by number

GET /customers/flat/{Number}

Description

Customers: Flat Customer Search by number

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Query	Number <i>required</i>	Number of customer to look for	string

Responses

HTTP Code	Description	Schema
200	Success	FlatCustomersResponse

Consumes

- `application/json`

Produces

- `application/json`

Tags

- customers

Security

Type	Name
basic	basic

Customers: Flat Customer Search by changed date

```
GET /customers/flat/{UpdatedAfter}/{CustomerNumbers}
```

Description

UpdatedAfter: Date after which an update took place. Attention: DateFormat: 'mm.dd.yyyy HH:mm:ss' or 'mm-dd-yyyy HH:mm:ss'. E.g.: <http://IP/customers/27.06.2016%2011:12:00/2.1.101>.
CustomerNumbers: List of (existing) HKS customer numbers in which the consumer is interested in.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Query	CustomerNumbers <i>required</i>	List of (existing) HKS customer numbers in which the consumer is interested in.	< string > array(multi)
Query	IncludeNewCustomers <i>optional</i>	If true, also newly created (not necessarily modified) customers will be returned.	boolean
Query	UpdatedAfter <i>required</i>	Date after which an update took place. DateFormat: 'mm.dd.yyyy HH:mm:ss' or 'mm-dd-yyyy HH:mm:ss'. E.g.: http://IP/customers/06.14.2016%2011:12:00/2.1.101	string (date-time)

Responses

HTTP Code	Description	Schema
200	Success	CustomersUpdatedAfterResponse

Consumes

- `application/json`

Produces

- `application/json`

Tags

- customers

Security

Type	Name
basic	basic

Customers: Customer Search by number

GET /customers/{Number}

Description

Customers: Customer Search by number

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Query	Number <i>required</i>	Customer number to search for	string

Responses

HTTP Code	Description	Schema
200	Success	CustomerResponse

Consumes

- `application/json`

Produces

- `application/json`

Tags

- customers

Security

Type	Name
basic	basic

Customers: Flat Customer Search by changed date

```
GET /customers/{UpdatedAfter}/{CustomerNumbers}
```

Description

Customers: Flat Customer Search by changed date

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Query	CustomerNumbers <i>required</i>	List of (existing) HKS customer numbers in which the consumer is interested in.	< string > array(multi)
Query	IncludeNewCustomers <i>optional</i>	If true, also newly created (not necessarily modified) customers will be returned.	boolean

Type	Name	Description	Schema
Query	UpdatedAfter <i>required</i>	Date after which an update took place. DateFormat: 'mm.dd.yyyy HH:mm:ss' or 'mm-dd-yyyy HH:mm:ss'. E.g.: http://IP/customers/06.14.2016%2011:12:00/2.1.101	string (date-time)

Responses

HTTP Code	Description	Schema
200	Success	CustomersUpdatedAfterResponse

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- customers

Security

Type	Name
basic	basic

EMobility: Post information about a charge session

POST /emobility/charged/

Description

Send a change session's data to the server.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Body	body <i>optional</i>		EMobilityChargeCompleted

Responses

HTTP Code	Description	Schema
200	Success	EMobilityChargeCompletedResponse

Consumes

- `application/json`

Produces

- `application/json`

Tags

- `emobility`

Security

Type	Name
basic	basic

EMobility: Query a medium's e-mobility functions

```
GET /emobility/{Medium}
```

Description

Query if a medium can activate a charge session.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Query	Device <i>optional</i>	The charging device's id.	string
Query	Medium <i>required</i>	The medium to be queried.	string

Responses

HTTP Code	Description	Schema
200	Success	EMobilityCheckMediumResponse

Consumes

- `application/json`

Produces

- `application/json`

Tags

- `emobility`

Security

Type	Name
basic	basic

EMobility: Query a medium's e-mobility functions

```
GET /emobility/{Medium}/{Device}
```

Description

Query if a medium can activate a charge session of a specific device.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Query	Device <i>optional</i>	The charging device's id.	string
Query	Medium <i>required</i>	The medium to be queried.	string

Responses

HTTP Code	Description	Schema
200	Success	EMobilityCheckMediumResponse

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- emobility

Security

Type	Name
basic	basic

Mediumdata: Create Medium

POST /mediumdata

Description

Creates a new (personalized if PersonId is given) medium in HKS if the given medium number does

not exist.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Body	body <i>optional</i>		NewMedium

Responses

HTTP Code	Description	Schema
200	Success	MediumDataResponse

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- [mediumdata](#)

Security

Type	Name
basic	basic

Mediumdata: Query open inhouse credit for medium

```
GET /mediumdata/inhousecredit/{Number}
```

Description

Query open inhouse credit for medium.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Query	Number <i>required</i>	The hardware serial or logical number of the medium to be queried.	string
Query	NumberType <i>optional</i>	'1' -> the number is to be interpreted as a hardware serial, all other values will be interpreted as logical medium numbers. If the type of the number is provided, database lookup will be noticeably faster.	integer (int16)

Responses

HTTP Code	Description	Schema
200	Success	InhouseCreditDataResponse

Consumes

- `application/json`

Produces

- `application/json`

Tags

- `mediumdata`

Security

Type	Name
basic	basic

Mediumdata: Query open inhouse credit for medium

```
GET /mediumdata/inhousecredit/{Number}/{NumberType}
```


Description

Query open inhouse credit for medium. NumberType '1' means the number is a hardware serial, all other values will be interpreted as logical medium numbers.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Query	Number <i>required</i>	The hardware serial or logical number of the medium to be queried.	string
Query	NumberType <i>optional</i>	'1' -> the number is to be interpreted as a hardware serial, all other values will be interpreted as logical medium numbers. If the type of the number is provided, database lookup will be noticeably faster.	integer (int16)

Responses

HTTP Code	Description	Schema
200	Success	InhouseCreditDataResponse

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- mediumdata

Security

Type	Name
basic	basic

Mediumdata: Lock a medium

```
PUT /mediumdata/lock/{Number}
```

Description

Locks an existing medium.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
FormData	Number <i>required</i>	The hardware serial or logical number of the medium to be locked.	string
FormData	NumberType <i>optional</i>	'1' -> the number is to be interpreted as a hardware serial, all other values will be interpreted as logical medium numbers. If the type of the number is provided, database lookup will be noticeably faster.	integer (int16)
Body	body <i>optional</i>		LockMedium

Responses

HTTP Code	Description	Schema
200	Success	LockMediumResponse

Consumes

- `application/x-www-form-urlencoded`

Produces

- `application/json`

Tags

- mediumdata

Security

Type	Name
basic	basic

Mediumdata: Lock a medium

```
PUT /mediumdata/lock/{Number}/{NumberType}
```

Description

Locks an existing medium. NumberType '1' means the number is a hardware serial, all other values will be interpreted as logical medium numbers.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
FormData	Number <i>required</i>	The hardware serial or logical number of the medium to be locked.	string
FormData	NumberType <i>optional</i>	'1' -> the number is to be interpreted as a hardware serial, all other values will be interpreted as logical medium numbers. If the type of the number is provided, database lookup will be noticeably faster.	integer (int16)
Body	body <i>optional</i>		LockMedium

Responses

HTTP Code	Description	Schema
200	Success	LockMediumResponse

Consumes

- `application/x-www-form-urlencoded`

Produces

- `application/json`

Tags

- `mediumdata`

Security

Type	Name
basic	basic

Mediumdata: Unlock a medium

```
PUT /mediumdata/unlock/{Number}
```

Description

Unlocks an existing medium.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
FormData	Number <i>required</i>	The hardware serial or logical number of the medium to be locked.	string
FormData	NumberType <i>optional</i>	'1' -> the number is to be interpreted as a hardware serial, all other values will be interpreted as logical medium numbers. If the type of the number is provided, database lookup will be noticeably faster.	integer (int16)
Body	body <i>optional</i>		UnlockMedium

Responses

HTTP Code	Description	Schema
200	Success	LockMediumResponse

Consumes

- `application/x-www-form-urlencoded`

Produces

- `application/json`

Tags

- `mediumdata`

Security

Type	Name
basic	basic

Mediumdata: Unlock a medium

```
PUT /mediumdata/unlock/{Number}/{NumberType}
```

Description

Unlocks an existing medium. NumberType '1' means the number is a hardware serial, all other values will be interpreted as logical medium numbers.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
FormData	Number <i>required</i>	The hardware serial or logical number of the medium to be locked.	string

Type	Name	Description	Schema
FormData	NumberType <i>optional</i>	'1' -> the number is to be interpreted as a hardware serial, all other values will be interpreted as logical medium numbers. If the type of the number is provided, database lookup will be noticeably faster.	integer (int16)
Body	body <i>optional</i>		UnlockMedium

Responses

HTTP Code	Description	Schema
200	Success	LockMediumResponse

Consumes

- `application/x-www-form-urlencoded`

Produces

- `application/json`

Tags

- `mediumdata`

Security

Type	Name
basic	basic

Mediumdata: Query value card data for medium

```
GET /mediumdata/valuecard/{Number}
```

Description

Query value card data for medium.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Query	From <i>optional</i>	Query value card data after this date	string (date-time)
Query	Number <i>required</i>	The hardware serial or logical number of the medium to be queried.	string
Query	NumberType <i>optional</i>	'1' -> the number is to be interpreted as a hardware serial, all other values will be interpreted as logical medium numbers. If the type of the number is provided, database lookup will be noticeably faster.	integer (int16)
Query	To <i>optional</i>	Query value card data before this date	string (date-time)

Responses

HTTP Code	Description	Schema
200	Success	ValueCardDataResponse

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- mediumdata

Security

Type	Name
basic	basic

Mediumdata: Query value card data for medium

```
GET /mediumdata/valuecard/{Number}/{NumberType}
```

Description

Query value card data for medium. NumberType '1' means the number is a hardware serial, all other values will be interpreted as logical medium numbers.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Query	From <i>optional</i>	Query value card data after this date	string (date-time)
Query	Number <i>required</i>	The hardware serial or logical number of the medium to be queried.	string
Query	NumberType <i>optional</i>	'1' -> the number is to be interpreted as a hardware serial, all other values will be interpreted as logical medium numbers. If the type of the number is provided, database lookup will be noticeably faster.	integer (int16)
Query	To <i>optional</i>	Query value card data before this date	string (date-time)

Responses

HTTP Code	Description	Schema
200	Success	ValueCardDataResponse

Consumes

- `application/json`

Produces

- `application/json`

Tags

- `mediumdata`

Security

Type	Name
basic	basic

Mediumdata: Query Medium Data

```
GET /mediumdata/{Number}
```

Description

IncludeMedia: fetch person picture (forces IncludePersonalization).

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Query	IncludeMedia <i>optional</i>	Include personalisation-relevant media like person picture.	boolean
Query	IncludePerson alization <i>optional</i>	Include the personalization of this medium, if any.	boolean
Query	Number <i>required</i>	The hardware serial or logical number of the medium to be queried.	string
Query	NumberType <i>optional</i>	'1' -> the number is to be interpreted as a hardware serial, all other values will be interpreted as logical medium numbers. If the type of the number is provided, database lookup will be noticeably faster.	integer (int16)

Type	Name	Description	Schema
Query	SellingruleFilter <i>optional</i>	Selling rule filter(s).	< integer (int32) > array(multi)
Query	ValidPerformancesOnly <i>optional</i>	Override default ValidPerformancesOnly behaviour. 'not set' -> result contains invalid performances if configured in service configuration, '0' -> result includes invalid performances, '1' -> result contains valid performances only	boolean

Responses

HTTP Code	Description	Schema
200	Success	MediumDataResponse

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- mediumdata

Security

Type	Name
basic	basic

Mediumdata: Query Medium Data

```
GET /mediumdata/{Number}/{NumberType}
```

Description

NumberType '1' means the number is a hardware serial, all other values will be interpreted as

logical medium numbers. IncludeMedia: fetch person picture (forces IncludePersonalization).

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Query	IncludeMedia <i>optional</i>	Include personalisation-relevant media like person picture.	boolean
Query	IncludePersonalization <i>optional</i>	Include the personalization of this medium, if any.	boolean
Query	Number <i>required</i>	The hardware serial or logical number of the medium to be queried.	string
Query	NumberType <i>optional</i>	'1' -> the number is to be interpreted as a hardware serial, all other values will be interpreted as logical medium numbers. If the type of the number is provided, database lookup will be noticeably faster.	integer (int16)
Query	SellingruleFilter <i>optional</i>	Selling rule filter(s).	< integer (int32) > array(multi)
Query	ValidPerformancesOnly <i>optional</i>	Override default ValidPerformancesOnly behaviour. 'not set' -> result contains invalid performances if configured in service configuration, '0' -> result includes invalid performances, '1' -> result contains valid performances only	boolean

Responses

HTTP Code	Description	Schema
200	Success	MediumDataResponse

Consumes

- `application/json`

Produces

- `application/json`

Tags

- `mediumdata`

Security

Type	Name
basic	basic

Partner: Query ExSI Partners

GET /partner

Description

Get ExSI partner information, e.g. list of all registered partners

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)

Responses

HTTP Code	Description	Schema
200	Success	PartnerResponse

Consumes

- `application/json`

Produces

- `application/json`

Tags

- partner

Security

Type	Name
basic	basic

POS: Cancel bookings

POST /pos/cancellation/bookings

Description

Cancels bookings.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Body	body <i>optional</i>		CancelBookings

Responses

HTTP Code	Description	Schema
200	Success	CancelResponse

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- pos

Security

Type	Name
basic	basic

POS: Cancel a receipt

POST /pos/cancellation/receipt

Description

Cancels a receipt.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Body	body <i>optional</i>		CancelReceipt

Responses

HTTP Code	Description	Schema
200	Success	CancelResponse

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- pos

Security

Type	Name
basic	basic

POS: Cancel a receipt (Deprecated)

POST /pos/cancelreceipt

Description

Cancels a receipt. Deprecated: Use /pos/cancellation/receipt instead.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Body	body <i>optional</i>		CancelReceiptOldRequest

Responses

HTTP Code	Description	Schema
200	Success	CancelReceiptResponse

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- pos

Security

Type	Name
basic	basic

POS: Route to get receipt information

```
GET /pos/receipt/{ReceiptNumber}
```

Description

Gets information about a receipt.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Path	ReceiptNumber <i>required</i>	Number of the receipt to be queried	string

Responses

HTTP Code	Description	Schema
200	Success	ReceiptResponse

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- pos

Security

Type	Name
basic	basic

POS: Sell a cart

POST /pos/sale

Description

Sells the items passed

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Body	body <i>optional</i>		SaleItems

Responses

HTTP Code	Description	Schema
200	Success	SaleItemsResponse

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- pos

Security

Type	Name
basic	basic

POS: Sell a cart (Deprecated)

POST /pos/saleitems

Description

Sale. Deprecated: Use /pos/sale instead.

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Body	body <i>optional</i>		SaleItemsOld

Responses

HTTP Code	Description	Schema
200	Success	SaleItemsResponse

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- pos

Security

Type	Name
basic	basic

Script: Execute a configured script

POST /script/exec

Description

Execute a configured script. The request body has to contains the script's parameters. E.g.: { "ScriptName": "MyScript", "MediumNumber": "123456789012345678901234", "LastName": "Mustermann", "FirstName": "Max", [...] }

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)
Body	body <i>optional</i>		ExecScriptRequest

Responses

HTTP Code	Description	Schema
200	Success	ExecScriptResponse

Consumes

- `application/json`

Produces

- `application/json`

Tags

- script

Security

Type	Name
basic	basic

Version: Request version information

GET /version

Description

Routes to get version information

Parameters

Type	Name	Description	Schema
Header	Accept <i>required</i>	Accept Header	enum (application/json)

Responses

HTTP Code	Description	Schema
200	Success	VersionResponse

Consumes

- [application/json](#)

Produces

- [application/json](#)

Tags

- version

Definitions

Booking

Booking

Name	Description	Schema
AlreadyUsed <i>optional</i>	Only for entrance articles: The entitlement sold with this booking has been used at least once.	boolean
Amount <i>optional</i>	Describes how many items of this article were sold	integer (int32)
ArticleNumber <i>required</i>		string
BookingMediums <i>optional</i>	List of mediums the article of this booking was sold on if applicable	< BookingMedium > array
CanBeCancelled <i>optional</i>	Describes if booking can be cancelled	boolean
IsCancellation <i>optional</i>	Describes if this booking is a cancellation of another booking	boolean
IsCancelled <i>optional</i>	Describes if booking has already been cancelled	boolean
Position <i>optional</i>	Position within the receipt	integer (int32)
ShortText <i>optional</i>	Short text describing the booking	string
TotalPrice <i>optional</i>	Total price of all units sold in this booking	number (double)
UnitPrice <i>optional</i>	Price of one unit	number (double)

BookingMedium

BookingMedium

Name	Description	Schema
HardwareSerial <i>optional</i>	Medium's physical id, e.g. barcode	string
Number <i>optional</i>	Logical medium number if applicable	integer (int32)
Type <i>optional</i>	Medium type: 0 = unknown/other; 1 = Mifare Classic; 2 = LegicPrime; 3 = Barcode; 4 = MagneticStrip; 5 = ISO15693; 6 = LegicAdvant	integer (Int16)

BookingToCancel

BookingToCancel

Name	Description	Schema
Amount <i>optional</i>	Optional amount of this booking to be cancelled. If not set, the complete booking is cancelled.	integer (int32)
Position <i>required</i>	Position of the booking on the receipt.	integer (int32)
ReceiptNumber <i>required</i>	Number of the receipt the booking belongs to.	string

CancelBookings

POS: Cancel bookings

Name	Description	Schema
BookingsToCancel <i>required</i>	List of bookings to be cancelled	< BookingToCancel > array
CancelDate <i>optional</i>	Date of this cancellation. If missing, current date is taken.	string (date-time)
Seller <i>optional</i>	Information about the seller, i.e. canceller in this case	Seller

CancelReceipt

POS: Cancel a receipt

Name	Description	Schema
CancelDate <i>optional</i>	Date of this cancellation. If missing, current date is taken. (not used in HKS)	string (date-time)
ReceiptNumber <i>required</i>	Number of the receipt to be cancelled.	string
Seller <i>optional</i>	Information about the seller. (not used in HKS)	Seller

CancelReceiptOldRequest

POS: Cancel a receipt (Deprecated)

Name	Schema
DeviceNumber <i>required</i>	integer (int32)
ReceiptNumber <i>required</i>	integer (int32)
ShiftNumber <i>required</i>	integer (int32)
Year <i>required</i>	integer (int16)

CancelReceiptResponse

CancelReceiptResponse

Name	Schema
ResponseStatus <i>optional</i>	ResponseStatus

CancelResponse

CancelResponse

Name	Description	Schema
CancelMode <i>required</i>	Cancel mode: 0 = receipt cancellation; 1 = booking cancellation.	integer (int16)
CanceledBookings <i>optional</i>	List of bookings cancelled.	< CanceledBooking > array
ReceiptNumber <i>required</i>	Receipt number created by the cancellation.	string
ResponseStatus <i>optional</i>		ResponseStatus

CanceledBooking

CanceledBooking

Name	Description	Schema
Amount <i>required</i>	Count of items of this article that were cancelled	integer (int32)
ArticleNumber <i>required</i>	Number of the article cancelled	string
OriginalAmount <i>required</i>	Count of items of this article that were originally sold	integer (int32)
Price <i>required</i>	Price of this cancel booking	number (double)
ShortText <i>required</i>	Text of the created cancel booking	string

Cart

Couponing: Get a cart's rebated prices

Name	Description	Schema
CartEntries <i>required</i>	List of cart entries to be sold.	< CartEntry > array
Code <i>required</i>	Coupon code	string

CartEntry

CartEntry

Name	Description	Schema
Amount <i>required</i>	Amount	integer (int32)
ArticleId <i>required</i>	ArticleId of the cart entry	string
Price <i>optional</i>	Total price of this cart entry. If no price is set, the service calculates the price itself.	number (double)

CartResponse

CartResponse

Name	Description	Schema
ArticleId <i>optional</i>	ID of the article associated with this coupon.	string
MultipleUsing <i>optional</i>	Coupon can be used more than once.	boolean
PersonalizationRequired <i>optional</i>	Coupon needs a customer.	boolean

Name	Description	Schema
RebateArticleFilter <i>optional</i>	Only if type = 1: filter for articles to be rebated.	string
RebateType <i>optional</i>	0 = relative, 1 = reduce unit price absolutely, 2 = reduce total price absolutely	integer (int32)
RebateValue <i>optional</i>	Only if type = 1: rebate factor in percent.	number (double)
RebatedCartEntries <i>optional</i>	List of cart entries with original and rebated prices.	< RebatedCartEntry > array
ResponseStatus <i>optional</i>		ResponseStatus
Type <i>optional</i>	Type of this coupon (1 = one time rebate, 2 = special article, 3 = special tariff).	integer (int32)
Valid <i>optional</i>	Validity of the coupon.	boolean

CatalogArticle

CatalogArticle

Name	Description	Schema
ConjugatedArticleIds <i>optional</i>	List of articles which are automatically sold with this article.	< ConjugatedArticle > array
Description <i>optional</i>	Text describing the article	string
Id <i>required</i>	Unique article id	string (uuid)
IsRebate <i>optional</i>	Indicates if an article is a rebate article	boolean

Name	Description	Schema
IsSubscription <i>optional</i>	Purchase of this article leads to a subscription	boolean
NeedsCustomer <i>optional</i>	Article sale needs customer information	boolean
NeedsMedium <i>optional</i>	If true, this article is sold to a medium.	boolean
Number <i>required</i>	Article number to be used in a sale request	string
OnlyOncePerMedium <i>optional</i>	If this article has to be sold to a medium, the medium must not contain more than one item on this article.	boolean
PaymentTypes <i>optional</i>	List of payment type this article can be purchased with: 0 = cash; 1 = invoice; 2 = direct debit; 3 = debit card; 4 = inhouse credit; 5 = value card; 6 = credit card	< integer (int32) > array
PriceType <i>required</i>	Price calculation type: 0 = normal (fix) price; 1 = null price; 3 = manual price; 4 = dynamically calculated	integer (int16)
RebateArticleFilter <i>optional</i>	Filter indicating which articles are rebated.	string
RebateType <i>optional</i>	Rebate type if applicable. 0 = relative, 1 = absolute on unit price, 2 = absolute on total price, 3 = relative with manual rebate, 4 = absolute with manual rebate	integer (int16)
RebateValue <i>optional</i>	Rebate amount	number (double)
SaleValidFrom <i>optional</i>	Earliest possible sale date. Null means not limited.	string (date-time)
SaleValidTo <i>optional</i>	Latest possible sale date (inclusive). Null means not limited.	string (date-time)

Name	Description	Schema
SalesRule <i>required</i>	Describes the kind of sales for this article: 1050 = shop article; 1060 = entrance article; 1080 = value card article; 1100 = cell reservation (e.g. a beach chair); 1110 = visitor tax; 1111 = contingent reservation (e.g. seminar place).	integer (int16)
ShortText <i>required</i>	Short article text	string
UnitPrice <i>optional</i>	Price of one unit (if determinable)	number (double)
VATRates <i>optional</i>	List of system's current vat rates	< number (double) > array
VATSplitting <i>required</i>	VAT calculation of this article. E.g.: "1=100" means 100% of the article price are subject to vat rate 1, "1=70;2=30" means 70% of the price are subject to vat rate 1 and 30% are subject to vat rate 2.	string
ValidDuration <i>optional</i>	Duration of validity if ValidityType is 0 oder 1. Format: YYYY.MM.DD.HH.MM.SS	string
ValidFrom <i>optional</i>	Begin of validity interval if ValidityType is 2. Null means not limited.	string (date-time)
ValidTo <i>optional</i>	End of validity interval if ValidityType is 2. Null means not limited.	string (date-time)
ValidityType <i>optional</i>	Describes how the validity of this article is determined if this article is sold to a medium. 0 : starts with first usage; 1: starts with sale; 2: absolute validity range (see ValidFrom and ValidTo); 3: variable (to be set with sale)	integer (int16)

CatalogRequest

Catalog: Request catalog information

Name	Description	Schema
DeviceID <i>optional</i>	Optional parameter specifying the device the catalog is meant for	string

CatalogResponse

CatalogResponse

Name	Description	Schema
CatalogArticles <i>required</i>	List of available articles	< CatalogArticle > array
CurrencyCode <i>required</i>	Currency of monetary values as numeric ISO 4217 code (e.g. 978 = EUR).	integer (int32)
ResponseStatus <i>optional</i>		ResponseStatus

ConjugatedArticle

ConjugatedArticle

Name	Description	Schema
Count <i>required</i>	Defines how many times this article is automatically sold	integer (int32)
Id <i>required</i>	Id of the automatically sold article	string (uuid)

Coupon

Couponing: Gets a coupon's definition

Name	Description	Schema
Code <i>required</i>	Coupon code	string

CouponResponse

CouponResponse

Name	Description	Schema
ArticleId <i>optional</i>	ID of the article associated with this coupon.	string
MultipleUsing <i>optional</i>	Coupon can be used more than once.	boolean
PersonalizationRequired <i>optional</i>	Coupon needs a customer.	boolean
RebateArticleFilter <i>optional</i>	Only if type = 1: filter for articles to be rebated.	string
RebateType <i>optional</i>	0 = relative, 1 = reduce unit price absolutely, 2 = reduce total price absolutely	integer (int32)
RebateValue <i>optional</i>	Only if type = 1: rebate factor in percent.	number (double)
ResponseStatus <i>optional</i>		ResponseStatus
Type <i>optional</i>	Type of this coupon (1 = one time rebate, 2 = special article, 3 = special tariff).	integer (int32)
Valid <i>optional</i>	Validity of the coupon.	boolean

CreateFlatCustomer

Customers: Flat Customer Creation

Name	Description	Schema
ClientId <i>required</i>	ClientId is the mapping-key (-> DMT:RequestMapping.Name) used for creating the customer with the correct operator and facility.	string
Customer <i>required</i>	The customer to be created. Operator and Facility for the new customer is automatically determined by the request's ClientId.	FlatCustomer

Customer

Couponing: Gets customer account coupons

Name	Description	Schema
CustomerId <i>required</i>	Coupon code	string

CustomerByNumber

Customers: Customer Search by number

Name	Description	Schema
Number <i>required</i>	Customer number to search for	string

CustomerCoupon

CustomerCoupon

Name	Description	Schema
ArticleId <i>optional</i>	ID of the article associated with this coupon.	string
MultipleUsing <i>optional</i>	Coupon can be used more than once.	boolean
PersonalizationRequired <i>optional</i>	Coupon needs a customer.	boolean
RebateArticleFilter <i>optional</i>	Only if type = 1: filter for articles to be rebated.	string
RebateType <i>optional</i>	0 = relative, 1 = reduce unit price absolutely, 2 = reduce total price absolutely	integer (int32)
RebateValue <i>optional</i>	Only if type = 1: rebate factor in percent.	number (double)

Name	Description	Schema
Type <i>optional</i>	Type of this coupon (1 = one time rebate, 2 = special article, 3 = special tariff).	integer (int32)
Valid <i>optional</i>	Validity of the coupon.	boolean

CustomerCouponResponse

CustomerCouponResponse

Name	Description	Schema
Coupons <i>optional</i>	List of coupons stored in the customer's account.	< CustomerCoupon > array
ResponseStatus <i>optional</i>		ResponseStatus

CustomerResponse

CustomerResponse

Name	Schema
Customer <i>required</i>	Customer
ResponseStatus <i>optional</i>	ResponseStatus

CustomersBySearch

Customers: Customer Search by properties

Name	Description	Schema
Birthdate <i>optional</i>		string (date-time)
City <i>optional</i>	City	string

Name	Description	Schema
FilterMainPerson <i>optional</i>	When set to true, returns customers with TheBear as the sole person. When set to false, a customer with all persons will be returned.	boolean
Firstname <i>optional</i>		string
Lastname <i>optional</i>	City	string
Mediumnumber <i>optional</i>		integer (int32)
Number <i>optional</i>		string
ResultLimit <i>optional</i>	Limits the number of returned results.	integer (int16)

CustomersResponse

CustomersResponse

Name	Schema
Customers <i>required</i>	< Customer > array
ResponseStatus <i>optional</i>	ResponseStatus

CustomersUpdatedAfter

Customers: Customer Search by changed date

Name	Description	Schema
CustomerNumbers <i>required</i>	List of (existing) HKS customer numbers in which the consumer is interested in.	< string > array

Name	Description	Schema
IncludeNewCustomers <i>optional</i>	If true, also newly created (not necessarily modified) customers will be returned.	boolean
UpdatedAfter <i>required</i>	Date after which an update took place. DateFormat: 'mm.dd.yyyy HH:mm:ss' or 'mm-dd-yyyy HH:mm:ss'. E.g.: http://IP/customers/06.14.2016%2011:12:00/2.1.101	string (date-time)

CustomersUpdatedAfterResponse

CustomersUpdatedAfterResponse

Name	Description	Schema
ResponseStatus <i>optional</i>		ResponseStatus
UpdatedCustomers <i>required</i>	List of updated customers	< Customer > array

Dictionary<String,String>

Dictionary<String,String>

Type : < string, string > map

EMobilityChargeCompleted

EMobility: Post information about a charge session

Name	Description	Schema
Counter <i>optional</i>	Charged energy in Wh	integer (int32)
Device <i>optional</i>	The charging device.	string
Duration <i>optional</i>	Charging time in seconds	integer (int32)

Name	Description	Schema
Medium <i>required</i>	The medium which charge session is sent.	string

EMobilityChargeCompletedResponse

EMobilityChargeCompletedResponse

Name	Schema
ResponseStatus <i>optional</i>	ResponseStatus
Success <i>optional</i>	boolean

EMobilityCheckMedium

EMobility: Query a medium's e-mobility functions

Name	Description	Schema
Device <i>optional</i>	The charging device's id.	string
Medium <i>required</i>	The medium to be queried.	string

EMobilityCheckMediumResponse

EMobilityCheckMediumResponse

Name	Description	Schema
CanCharge <i>optional</i>	True if charging is allowed.	boolean
ResponseStatus <i>optional</i>		ResponseStatus

ExecScriptRequest

Script: Execute a configured script

Name	Description	Schema
ScriptName <i>optional</i>	Name of the script to be executed	string

ExecScriptResponse

ExecScriptResponse

Name	Schema
ResponseStatus <i>optional</i>	ResponseStatus

FlatCustomer

FlatCustomer

Name	Description	Schema
AdvanceOrderMediumNumber <i>optional</i>	Readonly server-generated field. The number of a personalized medium if its cycle is not personalized.	string
AllowedPaymentTypes <i>optional</i>	Readonly server-generated field describing the payment types this customer may use.	< string > array
Birthdate <i>optional</i>		string (date-time)
City <i>required</i>	City	string
Comment <i>optional</i>	Customer comment	string
Company <i>optional</i>	Company	string
Country <i>optional</i>	Country	string

Name	Description	Schema
DebitorAccount <i>optional</i>	Readonly server-generated field.	string
Department <i>optional</i>	Department	string
DisabilityDegree <i>optional</i>	DisabilityDegree	string
DisabilityIdExpiry <i>optional</i>	Expiry date of DisabilityDegree	string (date-time)
Email <i>optional</i>	Email	string
Facility <i>optional</i>	Readonly server-generated field describing the facility this customer belongs to.	integer (int16)
Fax <i>optional</i>	Fax	string
Firstname <i>optional</i>	Firstname	string
Gender <i>optional</i>	0 = male; 1 = female, OBSOLETE. Use Salutation instead!	integer (int16)
Id <i>optional</i>	Customer number	string
IsTheBear <i>optional</i>	If more than one person belongs to this customer, this is the customer's main person.	boolean
Lastname <i>required</i>	Lastname	string
Locked <i>optional</i>	True means, customer is blocked.	boolean

Name	Description	Schema
Mobile <i>optional</i>	Mobile	string
NearTitle <i>optional</i>	NearTitle	string
Operator <i>optional</i>	Readonly server-generated field describing the operator this customer belongs to.	integer (int16)
PersonComment <i>optional</i>	Person comment	string
PersonId <i>optional</i>	Readonly server-generated field.	string (uuid)
Phone <i>optional</i>	Phone	string
Salutation <i>required</i>	0 = Mr./Herr; 1 = Mrs./Frau	integer (int16)
SepaMandate <i>optional</i>	SEPA mandate	SepaMandate
Street <i>optional</i>	Street	string
Title <i>optional</i>	Title	string
UpdatedAt <i>optional</i>	Readonly server-generated field. Last change date.	string (date-time)
Zip <i>optional</i>	Zip	string

FlatCustomerResponse

FlatCustomerResponse

Name	Description	Schema
Customer <i>optional</i>	Returned customer	FlatCustomer
ResponseStatus <i>optional</i>		ResponseStatus

FlatCustomersByNumber

Customers: Flat Customer Search by number

Name	Description	Schema
Number <i>required</i>	Number of customer to look for	string

FlatCustomersBySearch

Customers: Flat Customer Search by properties

Name	Description	Schema
Birthdate <i>optional</i>		string (date-time)
City <i>optional</i>		string
FilterMainPerson <i>optional</i>	When set to true, returns flat customers with TheBear as the sole person. When set to false, a flat customer for every person will be returned.	boolean
Firstname <i>optional</i>		string
Lastname <i>optional</i>		string
Mediumnumber <i>optional</i>		integer (int32)

Name	Description	Schema
Number <i>optional</i>		string
ResultLimit <i>optional</i>	Limits the number of returned results.	integer (int16)

FlatCustomersResponse

FlatCustomersResponse

Name	Description	Schema
Customers <i>optional</i>	Returned customers	< FlatCustomer > array
ResponseStatus <i>optional</i>		ResponseStatus

FlatCustomersUpdatedAfter

Customers: Flat Customer Search by changed date

Name	Description	Schema
CustomerNumbers <i>required</i>	List of (existing) HKS customer numbers in which the consumer is interested in.	< string > array
IncludeNewCustomers <i>optional</i>	If true, also newly created (not necessarily modified) customers will be returned.	boolean
UpdatedAfter <i>required</i>	Date after which an update took place. DateFormat: 'mm.dd.yyyy HH:mm:ss' or 'mm-dd-yyyy HH:mm:ss'. E.g.: http://IP/customers/06.14.2016%2011:12:00/2.1.101	string (date-time)

InhouseCreditBooking

InhouseCreditBooking

Name	Description	Schema
Amount <i>optional</i>	Describes how many items of this article were sold	integer (int32)
ArticleNumber <i>optional</i>	HKS article number	string
ShortText <i>optional</i>	Short text describing the booking	string
TotalPrice <i>optional</i>	Total price of all units sold in this booking	number (double)
UnitPrice <i>optional</i>	Price of one unit	number (double)

InhouseCreditData

Mediumdata: Query open inhouse credit for medium

Name	Description	Schema
Number <i>required</i>	The hardware serial or logical number of the medium to be queried.	string
NumberType <i>optional</i>	'1' -> the number is to be interpreted as a hardware serial, all other values will be interpreted as logical medium numbers. If the type of the number is provided, database lookup will be noticeably faster.	integer (int16)

InhouseCreditDataResponse

InhouseCreditDataResponse

Name	Description	Schema
InhouseCreditBookings <i>optional</i>	List of open inhouse credit bookings of this medium.	< InhouseCreditBooking > array
ResponseStatus <i>optional</i>		ResponseStatus

LockMedium

Mediumdata: Lock a medium

Name	Description	Schema
Number <i>required</i>	The hardware serial or logical number of the medium to be locked.	string
NumberType <i>optional</i>	'1' -> the number is to be interpreted as a hardware serial, all other values will be interpreted as logical medium numbers. If the type of the number is provided, database lookup will be noticeably faster.	integer (int16)

LockMediumResponse

LockMediumResponse

Name	Schema
ResponseStatus <i>optional</i>	ResponseStatus

Locker

Locker

Name	Description	Schema
Description <i>optional</i>	Locker description	string
Locked <i>optional</i>	True if locker is currently locked.	boolean
LockedAt <i>optional</i>	Date the Locker has been locked if applicable.	string (date-time)
LockerCategory <i>optional</i>	1 = Locker Box, 2 = Safe Deposit Box	integer (int32)
Number <i>optional</i>	Locker number	integer (int32)

Medium

Medium

Name	Description	Schema
CycleCreated <i>optional</i>	The date this medium's current cycle was created if applicable.	string (date-time)
CycleNumber <i>optional</i>	The number of this medium's current cycle if applicable.	integer (int32)
HardwareSerial <i>optional</i>	The hardware serial of this medium.	string
Lock <i>optional</i>	'0', if this medium is not locked.	integer (int32)
Lockers <i>optional</i>	Lockers used by medium	< Locker > array
Number <i>optional</i>	The logical number of this medium.	integer (int32)
Type <i>optional</i>	Type of this medium. This value is operator dependant!	integer (int32)
ValidTo <i>optional</i>	Expiry date of the validity of this medium, if any.	string (date-time)

MediumData

Mediumdata: Query Medium Data

Name	Description	Schema
IncludeMedia <i>optional</i>	Include personalisation-relevant media like person picture.	boolean
IncludePersonalization <i>optional</i>	Include the personalization of this medium, if any.	boolean

Name	Description	Schema
Number <i>required</i>	The hardware serial or logical number of the medium to be queried.	string
NumberType <i>optional</i>	'1' -> the number is to be interpreted as a hardware serial, all other values will be interpreted as logical medium numbers. If the type of the number is provided, database lookup will be noticeably faster.	integer (int16)
SellingruleFilter <i>optional</i>	Selling rule filter(s).	< integer (int32) > array
ValidPerformancesOnly <i>optional</i>	Override default ValidPerformancesOnly behaviour. 'not set' -> result contains invalid performances if configured in service configuration, '0' -> result includes invalid performances, '1' -> result contains valid performances only	boolean

MediumDataResponse

MediumDataResponse

Name	Description	Schema
InhouseCreditAmount <i>optional</i>	Amount of the medium's current cycle's inhouse credit.	number (double)
Medium <i>optional</i>	The medium DTO.	Medium
Owner <i>optional</i>	The owner of this medium, i.e. the personalization.	MediumOwner
Performances <i>optional</i>	List of available performances of this medium.	< Performance > array
ResponseStatus <i>optional</i>		ResponseStatus

Name	Description	Schema
ValueCardAmount <i>optional</i>	Amount of all value cards of the medium's current cycle.	number (double)

MediumOwner

MediumOwner

Name	Schema
City <i>optional</i>	string
CustomerName <i>optional</i>	string
DOB <i>optional</i>	string (date-time)
FirstName <i>optional</i>	string
Gender <i>optional</i>	integer (int32)
Id <i>optional</i>	string
LastName <i>optional</i>	string
Name <i>optional</i>	string
Number <i>optional</i>	string
Picture <i>optional</i>	string (binary)
Salutation <i>optional</i>	integer (int32)

Name	Schema
Street <i>optional</i>	string
Zip <i>optional</i>	string

NewMedium

Mediumdata: Create Medium

Name	Description	Schema
Number <i>required</i>	The number must be the hardware serial of the medium to be created.	string
PersonId <i>optional</i>	GUID of an existing person. The new medium will be personalized with this existing person.	string (uuid)

Object

Object

Type : object

OrderCoupon

OrderCoupon

Name	Description	Schema
Code <i>optional</i>	Coupon code to be printed via the output channel.	string
Picture <i>optional</i>	Base64 encoded picture accompanying this coupon (e.g. some promotional picture) to be printed via the output channel.	string (binary)

OrderCoupons

Couponing: Create and return coupons

Name	Description	Schema
ArticleIds <i>required</i>	List of articles that determine which coupons are returned.	< string > array
CustomerId <i>optional</i>	ID of the ordering customer.	string
GrandTotal <i>optional</i>	Grand Total of the order.	number (double)
OutputChannel <i>required</i>	Output channel: 1 = receipt, 2 = bill, 4 = delivery slip.	integer (int32)

OrderCouponsResponse

OrderCouponsResponse

Name	Description	Schema
Coupons <i>optional</i>	List of returned coupon-picture tuples	< OrderCoupon > array
ResponseStatus <i>optional</i>		ResponseStatus

Partner

Partner

Name	Description	Schema
Id <i>required</i>	Partner Id	integer (int32)
Name <i>required</i>	Partner name	string

PartnerRequest

Partner: Query ExSI Partners

Type : object

PartnerResponse

PartnerResponse

Name	Description	Schema
Partners <i>required</i>	List of all registered partners	< Partner > array
ResponseStatus <i>optional</i>		ResponseStatus

PaymentInfo

PaymentInfo

Name	Description	Schema
BrandName <i>optional</i>	Card brandname if debit or credit card payment	string
HardwareSerial <i>optional</i>	If no medium number is set: HKS medium's hardware for value card or inhouse credit to use	string
MediumNumber <i>optional</i>	HKS medium number for value card or inhouse credit to use	integer (int32)
PaymentType <i>required</i>	Type of this payment: 0 = cash; 1 = invoice (SaleItems.Contact is mandatory in this case); 2 = direct debit (SaleItems.Contact is mandatory in this case); 3 = debit card; 4 = inhouse credit; 5 = value card; 6 = credit card	integer (int16)

Performance

Performance

Name	Description	Schema
ActivatedAt <i>optional</i>		string (date-time)

Name	Description	Schema
ActivationValidTo <i>optional</i>		string (date-time)
Articlegroup <i>optional</i>	Textual representation of the article group.	string
Articlenumber <i>optional</i>	The full HKS article number. Structure: '<Operator>.<Facility>.<Group>.<Number>.<TariffGroup>.<Index>'.	string
BorderAmount <i>optional</i>		number (double)
ExternalArticleId <i>optional</i>		string
SellingRule <i>optional</i>		integer (int32)
Shorttext <i>optional</i>		string
UnitPrice <i>optional</i>		number (double)
UsedCount <i>optional</i>		number (double)
ValidFrom <i>optional</i>		string (date-time)
ValidTo <i>optional</i>	This date is to be interpreted as inclusive.	string (date-time)
VatCalculation <i>optional</i>	Rule used to calculate VAT, e.g. '1=70;2=30'.	string

RebatedCartEntry

RebatedCartEntry

Name	Description	Schema
Amount <i>optional</i>	Amount of the cart entry	integer (int32)
ArticleId <i>optional</i>	ArticleId of the cart entry	string
OriginalPrice <i>optional</i>	OriginalPrice	number (double)
RebatedPrice <i>optional</i>	RebatedPrice	number (double)

Receipt

POS: Route to get receipt information

Name	Description	Schema
ReceiptNumber <i>required</i>	Number of the receipt to be queried	string

ReceiptResponse

ReceiptResponse

Name	Description	Schema
Bookings <i>required</i>	List of bookings of this receipt	< Booking > array
CreationDate <i>required</i>	Creation date of this receipt	string (date-time)
CustomerNumber <i>optional</i>	Number of customer associated with the receipt if applicable.	string

Name	Description	Schema
ReceiptNumber <i>required</i>	Number of this receipt	string

RedeemCoupon

Couponing: Redeems a coupon

Name	Description	Schema
Code <i>required</i>	Coupon code.	string
CustomerId <i>optional</i>	ID of a customer associated with the coupon.	string
Date <i>optional</i>	Date of this process. If missing, current date is taken.	string (date-time)

RedeemCouponResponse

RedeemCouponResponse

Name	Schema
ResponseStatus <i>optional</i>	ResponseStatus

ResponseError

ResponseError

Name	Schema
ErrorCode <i>optional</i>	string
FieldName <i>optional</i>	string
Message <i>optional</i>	string

Name	Schema
Meta <i>optional</i>	Dictionary<String,String>

ResponseStatus

ResponseStatus

Name	Schema
ErrorCode <i>optional</i>	string
Errors <i>optional</i>	< ResponseError > array
Message <i>optional</i>	string
Meta <i>optional</i>	Dictionary<String,String>
StackTrace <i>optional</i>	string

SaleItem

SaleItem

Name	Description	Schema
Amount <i>required</i>	How many items of this article are to be sold?	integer (int32)
ArticleId <i>required</i>	Id of the HKS article to be sold	string
Customer <i>optional</i>	If the article needs a medium and this property is set, the medium is connected to this customer.	FlatCustomer
DeliveryMode <i>optional</i>	Describes how the article is delivered to the customer if applicable: 0 = pickup; 1 = email; 2 = postal mail; 3 = Skidata WebWallet. (not used in HKS)	integer (int32)

Name	Description	Schema
Id <i>optional</i>	Unique Id for this sale item. The SaleItemResponse can be identified with this property.	string (uuid)
MediumNumber <i>optional</i>	HKS medium number if this article is sold to an existing medium. Otherwise a new medium is created if the article needs one.	integer (int32)
PackageId <i>optional</i>	If the article needs a medium and this property is set, all SaleItems with the same PackageId are created on the same medium. If SaleItems with the same PackageId have different MediumNumbers, the operation will fail.	string (uuid)
Remarks <i>optional</i>	Optional remarks stored with this sale.	< string > array
ShortText <i>optional</i>	Text of the created booking, auto-generated with article text if not set	string
TotalPrice <i>optional</i>	Aggregated price of all sold units, auto-generated if not set	number (double)
UnitPrice <i>optional</i>	Price of one unit, auto-generated if not set	number (double)
ValidFrom <i>optional</i>	Begin of validity interval. Property has to be set if ValidityType of catalog article is 2.	string (date-time)
ValidTo <i>optional</i>	End of validity interval. Property has to be set if ValidityType of catalog article is 2.	string (date-time)

SaleItemResponse

SaleItemResponse

Name	Description	Schema
CustomerNumber <i>optional</i>	Customer number the medium is associated with if applicable.	string
ErrorMessage <i>optional</i>	Error message if sale failed.	string

Name	Description	Schema
HardwareSerial <i>optional</i>	UID (e.g. barcode) of the medium the article was created on if applicable.	string
MediumNumber <i>optional</i>	Logical number of the medium the article was created on if applicable.	integer (int32)
ReservationNumber <i>optional</i>	Reservation number the sale is associated with if applicable.	string
ResultCode <i>optional</i>	Sale result code: 0 = no error; any other = sale failed(tbd.)	integer (int32)

SaleItems

POS: Sell a cart

Name	Description	Schema
AllowPartialSale <i>optional</i>	If set, the interface tries to sell as many items as possible. If one item's sale fails, the process continues with the next item. If not set, the process fails if any item's sale fails. (not used in HKS)	boolean
Contact <i>optional</i>	Optional contact the sale is created for. If set, the receipt is connected to this contact.	FlatCustomer
ItemsToSale <i>optional</i>	Items to be sold	< SaleItem > array
PaymentInfos <i>required</i>	List of payment information	< PaymentInfo > array
SaleDate <i>optional</i>	Date of this sale. If missing, current date is taken.	string (date-time)
Seller <i>optional</i>	Information about the seller (not used in HKS)	Seller

SaleItemsOld

POS: Sell a cart (Deprecated)

Name	Description	Schema
AllowPartialSale <i>optional</i>	If set, the interface tries to sell as many items as possible. If one item's sale fails, the process continues with the next item. If not set, the process fails if any item's sale fails. (not used in HKS)	boolean
Contact <i>optional</i>	Optional contact the sale is created for. If set, the receipt is connected to this contact.	FlatCustomer
ItemsToSale <i>optional</i>	Items to be sold	< SaleItem > array
PaymentInfos <i>required</i>	List of payment information	< PaymentInfo > array
SaleDate <i>optional</i>	Date of this sale. If missing, current date is taken.	string (date-time)
Seller <i>optional</i>	Information about the seller (not used in HKS)	Seller

SaleItemsResponse

SaleItemsResponse

Name	Description	Schema
CustomerNumber <i>optional</i>	Customer number the receipt is associated with if applicable.	string
ReceiptNumber <i>required</i>	Created receipt number.	string
ResponseStatus <i>optional</i>		ResponseStatus

Name	Description	Schema
SaleItemResponses <i>required</i>	List of sale results for the SaleItems.	< SaleItemResponse > array

Seller

Seller

Name	Description	Schema
PosId <i>optional</i>	ID of the point-of-sale	string
PosName <i>optional</i>	Name of the point-of-sale	string
PosType <i>optional</i>	Type of the point-of-sale, e.g. webshop, cash desk, ticket machine, etc.	string
SellerId <i>optional</i>	ID of the seller	string
Sellername <i>optional</i>	Name of the seller	string

SepaMandate

SepaMandate

Name	Schema
Bic <i>optional</i>	string
Iban <i>optional</i>	string
Reference <i>optional</i>	string

UnlockMedium

Mediumdata: Unlock a medium

Name	Description	Schema
Number <i>required</i>	The hardware serial or logical number of the medium to be locked.	string
NumberType <i>optional</i>	'1' -> the number is to be interpreted as a hardware serial, all other values will be interpreted as logical medium numbers. If the type of the number is provided, database lookup will be noticeably faster.	integer (int16)

UpdateFlatCustomer

Customers: Flat Customer update

Name	Description	Schema
Customer <i>required</i>	Customer to update	FlatCustomer

ValueCardBooking

ValueCardBooking

Name	Description	Schema
SaleAmount <i>optional</i>	Describes how many items of this article were sold	integer (int32)
SaleArticleNumber <i>optional</i>	HKS article number	string
SaleShortText <i>optional</i>	Short text describing the booking	string
TimeStamp <i>optional</i>	Timestamp of this value card booking	string (date-time)
TotalPrice <i>optional</i>	Total price of all units sold in this booking	number (double)
Type <i>optional</i>	0 = value card pay-in, 1 = value card pay-out, 2 = sale by value card	number (Int32)

Name	Description	Schema
UnitPrice <i>optional</i>	Price of one unit	number (double)
ValueCardArticleNumber <i>optional</i>	HKS article number	string

ValueCardData

Mediumdata: Query value card data for medium

Name	Description	Schema
From <i>optional</i>	Query value card data after this date	string (date-time)
Number <i>required</i>	The hardware serial or logical number of the medium to be queried.	string
NumberType <i>optional</i>	'1' -> the number is to be interpreted as a hardware serial, all other values will be interpreted as logical medium numbers. If the type of the number is provided, database lookup will be noticeably faster.	integer (int16)
To <i>optional</i>	Query value card data before this date	string (date-time)

ValueCardDataResponse

ValueCardDataResponse

Name	Description	Schema
From <i>optional</i>	Only value card data after this date is returned.	string (date-time)
ResponseStatus <i>optional</i>		ResponseStatus
To <i>optional</i>	Only value card data before this date is returned.	string (date-time)

Name	Description	Schema
ValueCardBookings <i>optional</i>	List of value card bookings of this medium.	< ValueCardBooking > array

Version

Version: Request version information

Type : object

VersionResponse

VersionResponse

Name	Description	Schema
ApiVersion <i>optional</i>	Supported ExSI api version	string
ResponseStatus <i>optional</i>		ResponseStatus
Version <i>optional</i>	The web server's version information	string

Security

basic

Type : basic