**Hello World Template version: 1.0**

[[doc-title]]

[[doc-type]]

**Version: [[doc-version]]**

**Release Date: [[doc-release-date]]**

# Markdown1

## PUZZLES CLOUD MARKDOWN FILE HEADING 1

### Welcome to the puzzles cloud md file HEADING 2

#### Let's start with stupid jokes HEADING 3

##### **Let me tell you stupid joke on German: HEADING4**

* "Hey Philipp, wie war denn der Urlaub?"
* "Gräßlich! Im Hotel hatte ich Zimmernummer hundert. Und vom Türschild ist die Eins abgefallen!"

ÜÜÜÜÜÜ- that was stupid

##### Let's switch to the Czech language HEADING 4

* "Co rłńóobi trakłńótośźr u fśźryśźżzjera?"
* Warćękocze.

ąćęłńóśźż!!!!

**Just kidding, that was Poland language u fool, let's switch to the Czech for real**

###### Real Czech language HEADING 5

**ýžáčďéěíňóřšťúů**

* Jak říkáte albínskému muži z Prahy?
* Prázdný Čech.

## Longest Serbian word he he HEADING 1

Here you go: АБВГДЂЕЖЗИЈКЛЉМНЊОПРСТУФХЦЧЏШ

### Serbian literature HEADING 2

Весела песма  
 Из школе се враћа Стева  
 сав весео путем пева.  
   
 Уз пут среће бака Јелу:  
 „Ћирилицу знадем целу!"  
   
 Кући дође, па са врата:  
 „Ћирилицу знадем, тата!"  
   
 „Све сад, мајко, читам сам,  
 јер од данас и „ш" знам!"  
   
 Оком тражи и сестрицу:  
 „Сејо, знадем ћирилицу!"  
   
 Потом оде до икона,  
 па начини три поклона.  
   
 Хоће Стева да се јави  
 и захвали Светом Сави.  
   
 Свети Сава Стеву чује,  
 па се скупа с њим радује:  
   
 „Српство ми је од сад јаче  
 за још једно вредно ђаче!"

## It's enough, lets start with some smart things HEADING 1

1. PLAY THIS VIDEO AND THANKS ME LATER!

**HIT THAT PLAY BUTTON** 

Yep, it's a picture, fool... lol Here you go, CLICK IT:

# Markdown3

## Edge Service

The Edge Service provides the interface between the public REST API and internal AMQP API.

### TODO

#### TODO version 1.0

* [x] Test environment and credentials
* [x] specify voucher content fields
* [x] specify voucher pricing fields
* [x] specify payment method fields

#### TODO version 1.0-next

* [ ] refactor entities and apply relationships
* [ ] provide full resource schemas
* [ ] enforce validation of resource schemas

### Test environment and credentials

TODO

| Username | Password | Description |
| --- | --- | --- |
| [test@test.org](mailto:test@test.org) | test | just protected |
| [test@gmail.com](mailto:test@gmail.com) | test | just testing |
| testtest | test | qa |

### Edge Service Registry

The Edge Service Registry is responsible to maintain an index of all currently available Edge Services and configure all its known [Load-Balancers] accordingly.

As of writing no hosting solution has been chosen yet. Therefor the actual API which is used to configure the load-balancers remains unspecified for now.

All Edge Service Instances **must** register themselves on the registry on startup and un-register on shutdown.

### Edge Service Relay

Every Edge Service will be provided with a local relay connecting its instances to the central Message Broker cluster. The message exchange routing of the relay is configured in such a way, that only messages relevant to its connected Edge Services is routed to it from the broker cluster.

See [Message Relay] for more information.

### Edge Service Client

Each Service that is to be served by the Edge Service is represented by a so-called Edge Service Client. It is responsible to provide the implementation of the public REST API for that given service.

### ReST API

The Edge Service API acts as translation layer between the Services and the Frontend. The actual logic is implemented in separate [Edge Service Client]s for each individual Service consumed by the API or API endpoint provided by the API, respectively.

All public-facing endpoints **must** adhere to the [JSONAPI] specification and be subject to strict [Access Control].

#### Login

##### Login request

| Variables | Description | Example |
| --- | --- | --- |
| ICP\_SHOP\_ID | see [Test environment and credentials] |  |
| ICP\_SHOP\_PASSPHRASE | see [Test environment and credentials] |  |
| ICP\_SHOP\_DIGEST | Derived from ${ICP\_SHOP\_ID} and ${ICP\_SHOP\_PASSPHRASE} | MzAwMDAwMTY6dG9wLXNlY3JldC1wYXNzcGhyYXNl |

POST /api/rest/v1/auth/login HTTP/1.1  
 Authorization: Basic ${ICP\_SHOP\_DIGEST}  
 Accept: application/vnd.api+json

##### Login response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

##### Cookies

| Name | Value | Domain | Path | Expires | HttpOnly | Secure |
| --- | --- | --- | --- | --- | --- | --- |
| SESSION | MmQ5ZjkxOGItMTM4My00M2E0LWI1MDYtYmUwZjYwMzJiZDY5 | master.env.api.shop1.rocks | / | Seassion | true | false |

#### Logout

##### Logout request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

POST /api/rest/v1/auth/logout HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Logout response

HTTP/1.1 204 No Content

#### Voucher list

See [VoucherWebList] method in the [Mimas integration] documentation for the source of these fields.

##### Voucher list request

GET /api/rest/v1/vouchers HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher list response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data":[  
 {  
 "id":"f634f09f-9a06-4297-a326-0d703938b702",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL10",  
 "currency":"EUR",  
 "ean-code":"8717472242211",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 10 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 10",  
 "value":10.0,  
 "price-total":11.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods": {  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 }  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 }  
 }  
 },  
 {  
 "id":"a3473e20-19b4-44fe-8389-67a511854759",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL15",  
 "currency":"EUR",  
 "ean-code":"8717472242129",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 15 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 15",  
 "value":15.0,  
 "price-total":16.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods":[  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 }  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 ]  
 }  
 }  
 ]  
 }

#### Voucher detail

##### Voucher detail request

GET /api/rest/v1/vouchers/:voucher-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher detail response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data":{  
 "id":"f634f09f-9a06-4297-a326-0d703938b702",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL10",  
 "currency":"EUR",  
 "ean-code":"8717472242211",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 10 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 10",  
 "value":10.0,  
 "price-total":11.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods": {  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 },  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 }  
 }  
 }  
 }

#### Voucher order

See [VoucherWebReserve] method in the [Mimas integration] documentation for the source of these fields.

##### Voucher order request

| Variables | Description | Example | | ------------------------| -------------------------------------------- | --------------------------------------------------------------- | |

POST /api/rest/v1/voucher-orders HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "voucher-code": "VODADE15"  
 "payment-method": "Wirecard VISA",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "provider-text-url": "https://provider.texts",  
   
   
 "order-success-redirect-url": "/success",  
 "order-failed-redirect-url": "failed",  
 "order-canceled-redirect-url": "/cancel"  
   
 }  
 }  
 }

| field name | type | constrains | description | example |
| --- | --- | --- | --- | --- |
| id | UUID | globally unique | globally unique UUID | c91feaf3-5968-4b60-aa5b-62fe7fdd905c |
| type | string | n/a | type of request, must be "voucher-order", "voucher-order" |  |
| voucher-code | string | n/a | code of the voucher gathered from voucher list | "VODADE15" |
| payment-method | string | n/a | on of the allowed methods for the voucher, denoted in voucher list response | "Wirecard VISA" |
| notification-type | string | currently only "email" is supported | type of the notification target | "email" |
| notification-target | string | n/a | target adress | "[me@my-email.com](mailto:me@my-email.com)" |
| order-success-redirect-url | string | valid URL | Where the wirecard should redirect if payment was successfull | protect |
| order-failed-redirect-url | string | valid URL | Where the wirecard should redirect if payment fails | protect |
| order-canceled-redirect-url | string | valid URL | Where the wirecard should redirect if payment was canceled | protect |

##### Voucher order response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "notification-status": "pending",  
 "confirmation-status": "pending",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "payment-provider": "wirecard",  
 "payment-redirect-url": "https://wpp-test.wirecard.com/?wPaymentToken=y93bki675x2CdKP8-h0mLD2zDgf26ptjxSNRiKFiIlM",  
 "payment-session-id": "4b7c0aec-3db3-4cd2-9b22-4d41fb964fdc",  
 "payment-status": "pending",  
 "payment-method": "creditcard",  
 "voucher-code": "LEBALLINNL10",  
 "voucher-currency": "EUR",  
 "voucher-value": 10.0  
 "voucher-status": "pending",  
 "order-price-total": 11.99,  
 "order-tax-total": 0.00,  
 "order-fee-total": 0.99,  
 "order-status": "pending",  
 "serial-number": "1234",  
 "confirmation-date": "0",  
 "batch-number": "1234"  
 }  
 }  
 }

#### Voucher order detail

##### Voucher order detail request

GET /api/rest/v1/voucher-orders/:order-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher order detail response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "notification-status": "pending",  
 "confirmation-status": "pending",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "payment-provider": "wirecard",  
 "payment-redirect-url": "https://wpp-test.wirecard.com/?wPaymentToken=y93bki675x2CdKP8-h0mLD2zDgf26ptjxSNRiKFiIlM",  
 "payment-session-id": "4b7c0aec-3db3-4cd2-9b22-4d41fb964fdc",  
 "payment-status": "pending",  
 "payment-method": "creditcard",  
 "voucher-code": "LEBALLINNL10",  
 "voucher-currency": "EUR",  
 "voucher-value": 10.0  
 "voucher-status": "pending",  
 "order-price-total": 11.99,  
 "order-tax-total": 0.00,  
 "order-fee-total": 1.99,  
 "order-status": "pending",  
 "serial-number": "1234",  
 "confirmation-date": "1611313953175"  
 }  
 }  
 }

#### Provider Text

See [GetProvicerTexts] method in the [Mimas integration] documentation for the source of these fields.

##### Provider Text request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

GET /api/rest/v1/provider-texts/ HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

GET /api/rest/v1/provider-texts/:provider-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Provider Text response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

{  
 "data": [  
 {  
 "id": "f74bb480-d6df-47ad-8a17-8d7b1b176661",  
 "type": "voucher-text",  
 "attributes": {  
 "provider-code": "ABOUTYOU",  
 "kind": "title",  
 "title": "Produktname",  
 "body": "About You-Geschenkkarte"  
 }  
 },  
 {  
 "id": "a598bf3b-abe0-4fc4-80fa-135a84510c81",  
 "type": "voucher-text",  
 "attributes": {  
 "provider-code": "ABOUTYOU",  
 "kind": "subtitle",  
 "title": "Slogan",  
 "body": "Hol Dir per E-Mail eine About You-Geschenkkarte."  
 }  
 }  
 ]  
 }

#### Content

See [Content Service] for more information on the source of these fields.

##### Content request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

GET /api/rest/v1/content/i18next.json HTTP/1.1  
 Authorization: No Auth  
 Accept: application/json

##### Content response

HTTP/1.1 200 OK  
 Content-Type: application/json

{  
 "en": {  
 "world.hello": "Hello, World!",  
 "example.text": "Lorem ipsum dolor sit amet...",  
 "button.cancel": "Cancel",  
 "radio.options": {  
 "option-1": "Selection 1",  
 "option-2": "Selection 2",  
 }  
 },  
 "de": {  
 "world.hello": "Hallo, Welt!",  
 "example.text": "Lorem ipsum dolor sit amet...",  
 "button.cancel": "Abbrechen"  
 "radio.options": {  
 "option-1": "Auswahl 1",  
 "option-2": "Auswahl 2",  
 }  
 }  
 }

#### Verification

See for more information on the source of these fields.

##### Verification request by phone number

POST /api/rest/v1//auth/phone-number-verification HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": {  
 "id": "38f4dfef-9baa-4302-831f-e41df4ce4084",  
 "type": "phone-number-verification",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "sms-confirmation-code": "716569"  
 }  
 }  
 }

##### Verification by phone number response

HTTP/1.1 200 OK  
 Content-Type: application/json

{  
 "data": {  
 "type": "phone-number-verification",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "ok": false,  
 "remaining-number-of-attempts": 4,  
 "remaining-expiration-time": 0  
 }  
 }  
 }

##### Verification request by sms

POST /api/rest/v1/auth/sms-verification-code HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": {  
 "id": "48f4dfef-9baa-4302-831f-e41df4ce4084",  
 "type": "sms-verification-code",  
 "attributes": {  
 "phone-number": "+381691994655"  
 }  
 }  
 }

##### Verification by sms number response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "type": "sms-verification-code",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "ok": true,  
 "remaining-number-of-attempts": 2,  
 "remaining-delay-time": 20,  
 "remaining-expiration-time": 1800  
 }  
 }  
 }

#### Cookies

##### Cookies request

POST /api/rest/v1/cookies HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": [  
 {  
 "type": "cookie",  
 "id": "shopId",  
 "attributes": {  
 "name": "shopId",  
 "value": "1234"  
 }  
 },  
 {  
 "type": "cookie",  
 "id": "ns\_dev\_id",  
 "attributes": {  
 "name": "ns\_dev\_id",  
 "value": "device\_id"  
 }  
 }  
 ]  
 }

##### Cookies response

HTTP/1.1 200 OK

#### Documentation

Returns list of Json Schemas for request and response for requested rest endpoint. Use exact endpoint plus schema url parameter. For example, /api/rest/v1/vouchers?schema will return json schemas for /api/rest/v1/vouchers endpoint.

##### Documentation request

GET /api/rest/v1/vouchers?schema HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Documentation response

Complete Response is omitted because of size , here is just a part :

HTTP/1.1 200 OK

{  
 "required": [  
 "createdDate",  
 "notificationStatus",  
 "notificationTarget",  
 "notificationType",  
 "orderFeeTotal",  
 "orderPriceTotal",  
 "orderStatus",  
 "orderTaxTotal",  
 "paymentMethod",  
 "paymentProvider",  
 "paymentStatus",  
 "shortId",  
 "voucherCode",  
 "voucherCurrency",  
 "voucherStatus",  
 "voucherValue"  
 ]  
 }

#### i18 internationalization

* Retrieves i18next json based on requested language

##### i18 request

GET api/rest/v1/content/i18next.json/de HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

# Big

## Edge Service

123456 The Edge Service provides the interface between the public REST API and internal AMQP API.

### TODO

#### TODO version 1.0

* [x] Test environment and credentials
* [x] specify voucher content fields
* [x] specify voucher pricing fields
* [x] specify payment method fields

#### TODO version 1.0-next

* [ ] refactor entities and apply relationships
* [ ] provide full resource schemas
* [ ] enforce validation of resource schemas

### Test environment and credentials

TODO

| Username | Password | Description |
| --- | --- | --- |
| [test@test.org](mailto:test@test.org) | test | just protected |
| [test@gmail.com](mailto:test@gmail.com) | test | just testing |
| testtest | test | qa |

### Edge Service Registry

The Edge Service Registry is responsible to maintain an index of all currently available Edge Services and configure all its known [Load-Balancers] accordingly.

As of writing no hosting solution has been chosen yet. Therefor the actual API which is used to configure the load-balancers remains unspecified for now.

All Edge Service Instances **must** register themselves on the registry on startup and un-register on shutdown.

### Edge Service Relay

Every Edge Service will be provided with a local relay connecting its instances to the central Message Broker cluster. The message exchange routing of the relay is configured in such a way, that only messages relevant to its connected Edge Services is routed to it from the broker cluster.

See [Message Relay] for more information.

### Edge Service Client

Each Service that is to be served by the Edge Service is represented by a so-called Edge Service Client. It is responsible to provide the implementation of the public REST API for that given service.

### ReST API

The Edge Service API acts as translation layer between the Services and the Frontend. The actual logic is implemented in separate [Edge Service Client]s for each individual Service consumed by the API or API endpoint provided by the API, respectively.

All public-facing endpoints **must** adhere to the [JSONAPI] specification and be subject to strict [Access Control].

#### Login

##### Login request

| Variables | Description | Example |
| --- | --- | --- |
| ICP\_SHOP\_ID | see [Test environment and credentials] |  |
| ICP\_SHOP\_PASSPHRASE | see [Test environment and credentials] |  |
| ICP\_SHOP\_DIGEST | Derived from ${ICP\_SHOP\_ID} and ${ICP\_SHOP\_PASSPHRASE} | MzAwMDAwMTY6dG9wLXNlY3JldC1wYXNzcGhyYXNl |

POST /api/rest/v1/auth/login HTTP/1.1  
 Authorization: Basic ${ICP\_SHOP\_DIGEST}  
 Accept: application/vnd.api+json

##### Login response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

##### Cookies

| Name | Value | Domain | Path | Expires | HttpOnly | Secure |
| --- | --- | --- | --- | --- | --- | --- |
| SESSION | MmQ5ZjkxOGItMTM4My00M2E0LWI1MDYtYmUwZjYwMzJiZDY5 | master.env.api.shop1.rocks | / | Seassion | true | false |

#### Logout

##### Logout request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

POST /api/rest/v1/auth/logout HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Logout response

HTTP/1.1 204 No Content

#### Voucher list

See [VoucherWebList] method in the [Mimas integration] documentation for the source of these fields.

##### Voucher list request

GET /api/rest/v1/vouchers HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher list response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data":[  
 {  
 "id":"f634f09f-9a06-4297-a326-0d703938b702",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL10",  
 "currency":"EUR",  
 "ean-code":"8717472242211",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 10 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 10",  
 "value":10.0,  
 "price-total":11.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods": {  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 }  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 }  
 }  
 },  
 {  
 "id":"a3473e20-19b4-44fe-8389-67a511854759",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL15",  
 "currency":"EUR",  
 "ean-code":"8717472242129",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 15 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 15",  
 "value":15.0,  
 "price-total":16.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods":[  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 }  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 ]  
 }  
 }  
 ]  
 }

#### Voucher detail

##### Voucher detail request

GET /api/rest/v1/vouchers/:voucher-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher detail response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data":{  
 "id":"f634f09f-9a06-4297-a326-0d703938b702",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL10",  
 "currency":"EUR",  
 "ean-code":"8717472242211",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 10 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 10",  
 "value":10.0,  
 "price-total":11.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods": {  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 },  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 }  
 }  
 }  
 }

#### Voucher order

See [VoucherWebReserve] method in the [Mimas integration] documentation for the source of these fields.

##### Voucher order request

| Variables | Description | Example | | ------------------------| -------------------------------------------- | --------------------------------------------------------------- | |

POST /api/rest/v1/voucher-orders HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "voucher-code": "VODADE15"  
 "payment-method": "Wirecard VISA",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "provider-text-url": "https://provider.texts",  
   
   
 "order-success-redirect-url": "/success",  
 "order-failed-redirect-url": "failed",  
 "order-canceled-redirect-url": "/cancel"  
   
 }  
 }  
 }

| field name | type | constrains | description | example |
| --- | --- | --- | --- | --- |
| id | UUID | globally unique | globally unique UUID | c91feaf3-5968-4b60-aa5b-62fe7fdd905c |
| type | string | n/a | type of request, must be "voucher-order", "voucher-order" |  |
| voucher-code | string | n/a | code of the voucher gathered from voucher list | "VODADE15" |
| payment-method | string | n/a | on of the allowed methods for the voucher, denoted in voucher list response | "Wirecard VISA" |
| notification-type | string | currently only "email" is supported | type of the notification target | "email" |
| notification-target | string | n/a | target adress | "[me@my-email.com](mailto:me@my-email.com)" |
| order-success-redirect-url | string | valid URL | Where the wirecard should redirect if payment was successfull | protect |
| order-failed-redirect-url | string | valid URL | Where the wirecard should redirect if payment fails | protect |
| order-canceled-redirect-url | string | valid URL | Where the wirecard should redirect if payment was canceled | protect |

##### Voucher order response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "notification-status": "pending",  
 "confirmation-status": "pending",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "payment-provider": "wirecard",  
 "payment-redirect-url": "https://wpp-test.wirecard.com/?wPaymentToken=y93bki675x2CdKP8-h0mLD2zDgf26ptjxSNRiKFiIlM",  
 "payment-session-id": "4b7c0aec-3db3-4cd2-9b22-4d41fb964fdc",  
 "payment-status": "pending",  
 "payment-method": "creditcard",  
 "voucher-code": "LEBALLINNL10",  
 "voucher-currency": "EUR",  
 "voucher-value": 10.0  
 "voucher-status": "pending",  
 "order-price-total": 11.99,  
 "order-tax-total": 0.00,  
 "order-fee-total": 0.99,  
 "order-status": "pending",  
 "serial-number": "1234",  
 "confirmation-date": "0",  
 "batch-number": "1234"  
 }  
 }  
 }

#### Voucher order detail

##### Voucher order detail request

GET /api/rest/v1/voucher-orders/:order-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher order detail response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "notification-status": "pending",  
 "confirmation-status": "pending",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "payment-provider": "wirecard",  
 "payment-redirect-url": "https://wpp-test.wirecard.com/?wPaymentToken=y93bki675x2CdKP8-h0mLD2zDgf26ptjxSNRiKFiIlM",  
 "payment-session-id": "4b7c0aec-3db3-4cd2-9b22-4d41fb964fdc",  
 "payment-status": "pending",  
 "payment-method": "creditcard",  
 "voucher-code": "LEBALLINNL10",  
 "voucher-currency": "EUR",  
 "voucher-value": 10.0  
 "voucher-status": "pending",  
 "order-price-total": 11.99,  
 "order-tax-total": 0.00,  
 "order-fee-total": 1.99,  
 "order-status": "pending",  
 "serial-number": "1234",  
 "confirmation-date": "1611313953175"  
 }  
 }  
 }

#### Provider Text

See [GetProvicerTexts] method in the [Mimas integration] documentation for the source of these fields.

##### Provider Text request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

GET /api/rest/v1/provider-texts/ HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

GET /api/rest/v1/provider-texts/:provider-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Provider Text response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

{  
 "data": [  
 {  
 "id": "f74bb480-d6df-47ad-8a17-8d7b1b176661",  
 "type": "voucher-text",  
 "attributes": {  
 "provider-code": "ABOUTYOU",  
 "kind": "title",  
 "title": "Produktname",  
 "body": "About You-Geschenkkarte"  
 }  
 },  
 {  
 "id": "a598bf3b-abe0-4fc4-80fa-135a84510c81",  
 "type": "voucher-text",  
 "attributes": {  
 "provider-code": "ABOUTYOU",  
 "kind": "subtitle",  
 "title": "Slogan",  
 "body": "Hol Dir per E-Mail eine About You-Geschenkkarte."  
 }  
 }  
 ]  
 }

#### Content

See [Content Service] for more information on the source of these fields.

##### Content request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

GET /api/rest/v1/content/i18next.json HTTP/1.1  
 Authorization: No Auth  
 Accept: application/json

##### Content response

HTTP/1.1 200 OK  
 Content-Type: application/json

{  
 "en": {  
 "world.hello": "Hello, World!",  
 "example.text": "Lorem ipsum dolor sit amet...",  
 "button.cancel": "Cancel",  
 "radio.options": {  
 "option-1": "Selection 1",  
 "option-2": "Selection 2",  
 }  
 },  
 "de": {  
 "world.hello": "Hallo, Welt!",  
 "example.text": "Lorem ipsum dolor sit amet...",  
 "button.cancel": "Abbrechen"  
 "radio.options": {  
 "option-1": "Auswahl 1",  
 "option-2": "Auswahl 2",  
 }  
 }  
 }

#### Verification

See for more information on the source of these fields.

##### Verification request by phone number

POST /api/rest/v1//auth/phone-number-verification HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": {  
 "id": "38f4dfef-9baa-4302-831f-e41df4ce4084",  
 "type": "phone-number-verification",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "sms-confirmation-code": "716569"  
 }  
 }  
 }

##### Verification by phone number response

HTTP/1.1 200 OK  
 Content-Type: application/json

{  
 "data": {  
 "type": "phone-number-verification",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "ok": false,  
 "remaining-number-of-attempts": 4,  
 "remaining-expiration-time": 0  
 }  
 }  
 }

##### Verification request by sms

POST /api/rest/v1/auth/sms-verification-code HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": {  
 "id": "48f4dfef-9baa-4302-831f-e41df4ce4084",  
 "type": "sms-verification-code",  
 "attributes": {  
 "phone-number": "+381691994655"  
 }  
 }  
 }

##### Verification by sms number response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "type": "sms-verification-code",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "ok": true,  
 "remaining-number-of-attempts": 2,  
 "remaining-delay-time": 20,  
 "remaining-expiration-time": 1800  
 }  
 }  
 }

#### Cookies

##### Cookies request

POST /api/rest/v1/cookies HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": [  
 {  
 "type": "cookie",  
 "id": "shopId",  
 "attributes": {  
 "name": "shopId",  
 "value": "1234"  
 }  
 },  
 {  
 "type": "cookie",  
 "id": "ns\_dev\_id",  
 "attributes": {  
 "name": "ns\_dev\_id",  
 "value": "device\_id"  
 }  
 }  
 ]  
 }

##### Cookies response

HTTP/1.1 200 OK

#### Documentation

Returns list of Json Schemas for request and response for requested rest endpoint. Use exact endpoint plus schema url parameter. For example, /api/rest/v1/vouchers?schema will return json schemas for /api/rest/v1/vouchers endpoint.

##### Documentation request

GET /api/rest/v1/vouchers?schema HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Documentation response

Complete Response is omitted because of size , here is just a part :

HTTP/1.1 200 OK

{  
 "required": [  
 "createdDate",  
 "notificationStatus",  
 "notificationTarget",  
 "notificationType",  
 "orderFeeTotal",  
 "orderPriceTotal",  
 "orderStatus",  
 "orderTaxTotal",  
 "paymentMethod",  
 "paymentProvider",  
 "paymentStatus",  
 "shortId",  
 "voucherCode",  
 "voucherCurrency",  
 "voucherStatus",  
 "voucherValue"  
 ]  
 }

#### i18 internationalization

* Retrieves i18next json based on requested language

##### i18 request

GET api/rest/v1/content/i18next.json/de HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

## Edge Service

The Edge Service provides the interface between the public REST API and internal AMQP API.

### TODO

#### TODO version 1.0

* [x] Test environment and credentials
* [x] specify voucher content fields
* [x] specify voucher pricing fields
* [x] specify payment method fields

#### TODO version 1.0-next

* [ ] refactor entities and apply relationships
* [ ] provide full resource schemas
* [ ] enforce validation of resource schemas

### Test environment and credentials

TODO

| Username | Password | Description |
| --- | --- | --- |
| [test@test.org](mailto:test@test.org) | test | just protected |
| [test@gmail.com](mailto:test@gmail.com) | test | just testing |
| testtest | test | qa |

### Edge Service Registry

The Edge Service Registry is responsible to maintain an index of all currently available Edge Services and configure all its known [Load-Balancers] accordingly.

As of writing no hosting solution has been chosen yet. Therefor the actual API which is used to configure the load-balancers remains unspecified for now.

All Edge Service Instances **must** register themselves on the registry on startup and un-register on shutdown.

### Edge Service Relay

Every Edge Service will be provided with a local relay connecting its instances to the central Message Broker cluster. The message exchange routing of the relay is configured in such a way, that only messages relevant to its connected Edge Services is routed to it from the broker cluster.

See [Message Relay] for more information.

### Edge Service Client

Each Service that is to be served by the Edge Service is represented by a so-called Edge Service Client. It is responsible to provide the implementation of the public REST API for that given service.

### ReST API

The Edge Service API acts as translation layer between the Services and the Frontend. The actual logic is implemented in separate [Edge Service Client]s for each individual Service consumed by the API or API endpoint provided by the API, respectively.

All public-facing endpoints **must** adhere to the [JSONAPI] specification and be subject to strict [Access Control].

#### Login

##### Login request

| Variables | Description | Example |
| --- | --- | --- |
| ICP\_SHOP\_ID | see [Test environment and credentials] |  |
| ICP\_SHOP\_PASSPHRASE | see [Test environment and credentials] |  |
| ICP\_SHOP\_DIGEST | Derived from ${ICP\_SHOP\_ID} and ${ICP\_SHOP\_PASSPHRASE} | MzAwMDAwMTY6dG9wLXNlY3JldC1wYXNzcGhyYXNl |

POST /api/rest/v1/auth/login HTTP/1.1  
 Authorization: Basic ${ICP\_SHOP\_DIGEST}  
 Accept: application/vnd.api+json

##### Login response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

##### Cookies

| Name | Value | Domain | Path | Expires | HttpOnly | Secure |
| --- | --- | --- | --- | --- | --- | --- |
| SESSION | MmQ5ZjkxOGItMTM4My00M2E0LWI1MDYtYmUwZjYwMzJiZDY5 | master.env.api.shop1.rocks | / | Seassion | true | false |

#### Logout

##### Logout request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

POST /api/rest/v1/auth/logout HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Logout response

HTTP/1.1 204 No Content

#### Voucher list

See [VoucherWebList] method in the [Mimas integration] documentation for the source of these fields.

##### Voucher list request

GET /api/rest/v1/vouchers HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher list response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data":[  
 {  
 "id":"f634f09f-9a06-4297-a326-0d703938b702",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL10",  
 "currency":"EUR",  
 "ean-code":"8717472242211",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 10 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 10",  
 "value":10.0,  
 "price-total":11.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods": {  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 }  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 }  
 }  
 },  
 {  
 "id":"a3473e20-19b4-44fe-8389-67a511854759",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL15",  
 "currency":"EUR",  
 "ean-code":"8717472242129",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 15 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 15",  
 "value":15.0,  
 "price-total":16.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods":[  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 }  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 ]  
 }  
 }  
 ]  
 }

#### Voucher detail

##### Voucher detail request

GET /api/rest/v1/vouchers/:voucher-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher detail response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data":{  
 "id":"f634f09f-9a06-4297-a326-0d703938b702",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL10",  
 "currency":"EUR",  
 "ean-code":"8717472242211",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 10 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 10",  
 "value":10.0,  
 "price-total":11.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods": {  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 },  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 }  
 }  
 }  
 }

#### Voucher order

See [VoucherWebReserve] method in the [Mimas integration] documentation for the source of these fields.

##### Voucher order request

| Variables | Description | Example | | ------------------------| -------------------------------------------- | --------------------------------------------------------------- | |

POST /api/rest/v1/voucher-orders HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "voucher-code": "VODADE15"  
 "payment-method": "Wirecard VISA",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "provider-text-url": "https://provider.texts",  
   
   
 "order-success-redirect-url": "/success",  
 "order-failed-redirect-url": "failed",  
 "order-canceled-redirect-url": "/cancel"  
   
 }  
 }  
 }

| field name | type | constrains | description | example |
| --- | --- | --- | --- | --- |
| id | UUID | globally unique | globally unique UUID | c91feaf3-5968-4b60-aa5b-62fe7fdd905c |
| type | string | n/a | type of request, must be "voucher-order", "voucher-order" |  |
| voucher-code | string | n/a | code of the voucher gathered from voucher list | "VODADE15" |
| payment-method | string | n/a | on of the allowed methods for the voucher, denoted in voucher list response | "Wirecard VISA" |
| notification-type | string | currently only "email" is supported | type of the notification target | "email" |
| notification-target | string | n/a | target adress | "[me@my-email.com](mailto:me@my-email.com)" |
| order-success-redirect-url | string | valid URL | Where the wirecard should redirect if payment was successfull | protect |
| order-failed-redirect-url | string | valid URL | Where the wirecard should redirect if payment fails | protect |
| order-canceled-redirect-url | string | valid URL | Where the wirecard should redirect if payment was canceled | protect |

##### Voucher order response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "notification-status": "pending",  
 "confirmation-status": "pending",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "payment-provider": "wirecard",  
 "payment-redirect-url": "https://wpp-test.wirecard.com/?wPaymentToken=y93bki675x2CdKP8-h0mLD2zDgf26ptjxSNRiKFiIlM",  
 "payment-session-id": "4b7c0aec-3db3-4cd2-9b22-4d41fb964fdc",  
 "payment-status": "pending",  
 "payment-method": "creditcard",  
 "voucher-code": "LEBALLINNL10",  
 "voucher-currency": "EUR",  
 "voucher-value": 10.0  
 "voucher-status": "pending",  
 "order-price-total": 11.99,  
 "order-tax-total": 0.00,  
 "order-fee-total": 0.99,  
 "order-status": "pending",  
 "serial-number": "1234",  
 "confirmation-date": "0",  
 "batch-number": "1234"  
 }  
 }  
 }

#### Voucher order detail

##### Voucher order detail request

GET /api/rest/v1/voucher-orders/:order-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher order detail response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "notification-status": "pending",  
 "confirmation-status": "pending",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "payment-provider": "wirecard",  
 "payment-redirect-url": "https://wpp-test.wirecard.com/?wPaymentToken=y93bki675x2CdKP8-h0mLD2zDgf26ptjxSNRiKFiIlM",  
 "payment-session-id": "4b7c0aec-3db3-4cd2-9b22-4d41fb964fdc",  
 "payment-status": "pending",  
 "payment-method": "creditcard",  
 "voucher-code": "LEBALLINNL10",  
 "voucher-currency": "EUR",  
 "voucher-value": 10.0  
 "voucher-status": "pending",  
 "order-price-total": 11.99,  
 "order-tax-total": 0.00,  
 "order-fee-total": 1.99,  
 "order-status": "pending",  
 "serial-number": "1234",  
 "confirmation-date": "1611313953175"  
 }  
 }  
 }

#### Provider Text

See [GetProvicerTexts] method in the [Mimas integration] documentation for the source of these fields.

##### Provider Text request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

GET /api/rest/v1/provider-texts/ HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

GET /api/rest/v1/provider-texts/:provider-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Provider Text response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

{  
 "data": [  
 {  
 "id": "f74bb480-d6df-47ad-8a17-8d7b1b176661",  
 "type": "voucher-text",  
 "attributes": {  
 "provider-code": "ABOUTYOU",  
 "kind": "title",  
 "title": "Produktname",  
 "body": "About You-Geschenkkarte"  
 }  
 },  
 {  
 "id": "a598bf3b-abe0-4fc4-80fa-135a84510c81",  
 "type": "voucher-text",  
 "attributes": {  
 "provider-code": "ABOUTYOU",  
 "kind": "subtitle",  
 "title": "Slogan",  
 "body": "Hol Dir per E-Mail eine About You-Geschenkkarte."  
 }  
 }  
 ]  
 }

#### Content

See [Content Service] for more information on the source of these fields.

##### Content request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

GET /api/rest/v1/content/i18next.json HTTP/1.1  
 Authorization: No Auth  
 Accept: application/json

##### Content response

HTTP/1.1 200 OK  
 Content-Type: application/json

{  
 "en": {  
 "world.hello": "Hello, World!",  
 "example.text": "Lorem ipsum dolor sit amet...",  
 "button.cancel": "Cancel",  
 "radio.options": {  
 "option-1": "Selection 1",  
 "option-2": "Selection 2",  
 }  
 },  
 "de": {  
 "world.hello": "Hallo, Welt!",  
 "example.text": "Lorem ipsum dolor sit amet...",  
 "button.cancel": "Abbrechen"  
 "radio.options": {  
 "option-1": "Auswahl 1",  
 "option-2": "Auswahl 2",  
 }  
 }  
 }

#### Verification

See for more information on the source of these fields.

##### Verification request by phone number

POST /api/rest/v1//auth/phone-number-verification HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": {  
 "id": "38f4dfef-9baa-4302-831f-e41df4ce4084",  
 "type": "phone-number-verification",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "sms-confirmation-code": "716569"  
 }  
 }  
 }

##### Verification by phone number response

HTTP/1.1 200 OK  
 Content-Type: application/json

{  
 "data": {  
 "type": "phone-number-verification",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "ok": false,  
 "remaining-number-of-attempts": 4,  
 "remaining-expiration-time": 0  
 }  
 }  
 }

##### Verification request by sms

POST /api/rest/v1/auth/sms-verification-code HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": {  
 "id": "48f4dfef-9baa-4302-831f-e41df4ce4084",  
 "type": "sms-verification-code",  
 "attributes": {  
 "phone-number": "+381691994655"  
 }  
 }  
 }

##### Verification by sms number response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "type": "sms-verification-code",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "ok": true,  
 "remaining-number-of-attempts": 2,  
 "remaining-delay-time": 20,  
 "remaining-expiration-time": 1800  
 }  
 }  
 }

#### Cookies

##### Cookies request

POST /api/rest/v1/cookies HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": [  
 {  
 "type": "cookie",  
 "id": "shopId",  
 "attributes": {  
 "name": "shopId",  
 "value": "1234"  
 }  
 },  
 {  
 "type": "cookie",  
 "id": "ns\_dev\_id",  
 "attributes": {  
 "name": "ns\_dev\_id",  
 "value": "device\_id"  
 }  
 }  
 ]  
 }

##### Cookies response

HTTP/1.1 200 OK

#### Documentation

Returns list of Json Schemas for request and response for requested rest endpoint. Use exact endpoint plus schema url parameter. For example, /api/rest/v1/vouchers?schema will return json schemas for /api/rest/v1/vouchers endpoint.

##### Documentation request

GET /api/rest/v1/vouchers?schema HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Documentation response

Complete Response is omitted because of size , here is just a part :

HTTP/1.1 200 OK

{  
 "required": [  
 "createdDate",  
 "notificationStatus",  
 "notificationTarget",  
 "notificationType",  
 "orderFeeTotal",  
 "orderPriceTotal",  
 "orderStatus",  
 "orderTaxTotal",  
 "paymentMethod",  
 "paymentProvider",  
 "paymentStatus",  
 "shortId",  
 "voucherCode",  
 "voucherCurrency",  
 "voucherStatus",  
 "voucherValue"  
 ]  
 }

#### i18 internationalization

* Retrieves i18next json based on requested language

##### i18 request

GET api/rest/v1/content/i18next.json/de HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

## Edge Service

The Edge Service provides the interface between the public REST API and internal AMQP API.

### TODO

#### TODO version 1.0

* [x] Test environment and credentials
* [x] specify voucher content fields
* [x] specify voucher pricing fields
* [x] specify payment method fields

#### TODO version 1.0-next

* [ ] refactor entities and apply relationships
* [ ] provide full resource schemas
* [ ] enforce validation of resource schemas

### Test environment and credentials

TODO

| Username | Password | Description |
| --- | --- | --- |
| [test@test.org](mailto:test@test.org) | test | just protected |
| [test@gmail.com](mailto:test@gmail.com) | test | just testing |
| testtest | test | qa |

### Edge Service Registry

The Edge Service Registry is responsible to maintain an index of all currently available Edge Services and configure all its known [Load-Balancers] accordingly.

As of writing no hosting solution has been chosen yet. Therefor the actual API which is used to configure the load-balancers remains unspecified for now.

All Edge Service Instances **must** register themselves on the registry on startup and un-register on shutdown.

### Edge Service Relay

Every Edge Service will be provided with a local relay connecting its instances to the central Message Broker cluster. The message exchange routing of the relay is configured in such a way, that only messages relevant to its connected Edge Services is routed to it from the broker cluster.

See [Message Relay] for more information.

### Edge Service Client

Each Service that is to be served by the Edge Service is represented by a so-called Edge Service Client. It is responsible to provide the implementation of the public REST API for that given service.

### ReST API

The Edge Service API acts as translation layer between the Services and the Frontend. The actual logic is implemented in separate [Edge Service Client]s for each individual Service consumed by the API or API endpoint provided by the API, respectively.

All public-facing endpoints **must** adhere to the [JSONAPI] specification and be subject to strict [Access Control].

#### Login

##### Login request

| Variables | Description | Example |
| --- | --- | --- |
| ICP\_SHOP\_ID | see [Test environment and credentials] |  |
| ICP\_SHOP\_PASSPHRASE | see [Test environment and credentials] |  |
| ICP\_SHOP\_DIGEST | Derived from ${ICP\_SHOP\_ID} and ${ICP\_SHOP\_PASSPHRASE} | MzAwMDAwMTY6dG9wLXNlY3JldC1wYXNzcGhyYXNl |

POST /api/rest/v1/auth/login HTTP/1.1  
 Authorization: Basic ${ICP\_SHOP\_DIGEST}  
 Accept: application/vnd.api+json

##### Login response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

##### Cookies

| Name | Value | Domain | Path | Expires | HttpOnly | Secure |
| --- | --- | --- | --- | --- | --- | --- |
| SESSION | MmQ5ZjkxOGItMTM4My00M2E0LWI1MDYtYmUwZjYwMzJiZDY5 | master.env.api.shop1.rocks | / | Seassion | true | false |

#### Logout

##### Logout request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

POST /api/rest/v1/auth/logout HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Logout response

HTTP/1.1 204 No Content

#### Voucher list

See [VoucherWebList] method in the [Mimas integration] documentation for the source of these fields.

##### Voucher list request

GET /api/rest/v1/vouchers HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher list response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data":[  
 {  
 "id":"f634f09f-9a06-4297-a326-0d703938b702",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL10",  
 "currency":"EUR",  
 "ean-code":"8717472242211",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 10 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 10",  
 "value":10.0,  
 "price-total":11.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods": {  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 }  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 }  
 }  
 },  
 {  
 "id":"a3473e20-19b4-44fe-8389-67a511854759",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL15",  
 "currency":"EUR",  
 "ean-code":"8717472242129",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 15 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 15",  
 "value":15.0,  
 "price-total":16.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods":[  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 }  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 ]  
 }  
 }  
 ]  
 }

#### Voucher detail

##### Voucher detail request

GET /api/rest/v1/vouchers/:voucher-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher detail response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data":{  
 "id":"f634f09f-9a06-4297-a326-0d703938b702",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL10",  
 "currency":"EUR",  
 "ean-code":"8717472242211",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 10 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 10",  
 "value":10.0,  
 "price-total":11.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods": {  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 },  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 }  
 }  
 }  
 }

#### Voucher order

See [VoucherWebReserve] method in the [Mimas integration] documentation for the source of these fields.

##### Voucher order request

| Variables | Description | Example | | ------------------------| -------------------------------------------- | --------------------------------------------------------------- | |

POST /api/rest/v1/voucher-orders HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "voucher-code": "VODADE15"  
 "payment-method": "Wirecard VISA",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "provider-text-url": "https://provider.texts",  
   
   
 "order-success-redirect-url": "/success",  
 "order-failed-redirect-url": "failed",  
 "order-canceled-redirect-url": "/cancel"  
   
 }  
 }  
 }

| field name | type | constrains | description | example |
| --- | --- | --- | --- | --- |
| id | UUID | globally unique | globally unique UUID | c91feaf3-5968-4b60-aa5b-62fe7fdd905c |
| type | string | n/a | type of request, must be "voucher-order", "voucher-order" |  |
| voucher-code | string | n/a | code of the voucher gathered from voucher list | "VODADE15" |
| payment-method | string | n/a | on of the allowed methods for the voucher, denoted in voucher list response | "Wirecard VISA" |
| notification-type | string | currently only "email" is supported | type of the notification target | "email" |
| notification-target | string | n/a | target adress | "[me@my-email.com](mailto:me@my-email.com)" |
| order-success-redirect-url | string | valid URL | Where the wirecard should redirect if payment was successfull | protect |
| order-failed-redirect-url | string | valid URL | Where the wirecard should redirect if payment fails | protect |
| order-canceled-redirect-url | string | valid URL | Where the wirecard should redirect if payment was canceled | protect |

##### Voucher order response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "notification-status": "pending",  
 "confirmation-status": "pending",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "payment-provider": "wirecard",  
 "payment-redirect-url": "https://wpp-test.wirecard.com/?wPaymentToken=y93bki675x2CdKP8-h0mLD2zDgf26ptjxSNRiKFiIlM",  
 "payment-session-id": "4b7c0aec-3db3-4cd2-9b22-4d41fb964fdc",  
 "payment-status": "pending",  
 "payment-method": "creditcard",  
 "voucher-code": "LEBALLINNL10",  
 "voucher-currency": "EUR",  
 "voucher-value": 10.0  
 "voucher-status": "pending",  
 "order-price-total": 11.99,  
 "order-tax-total": 0.00,  
 "order-fee-total": 0.99,  
 "order-status": "pending",  
 "serial-number": "1234",  
 "confirmation-date": "0",  
 "batch-number": "1234"  
 }  
 }  
 }

#### Voucher order detail

##### Voucher order detail request

GET /api/rest/v1/voucher-orders/:order-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher order detail response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "notification-status": "pending",  
 "confirmation-status": "pending",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "payment-provider": "wirecard",  
 "payment-redirect-url": "https://wpp-test.wirecard.com/?wPaymentToken=y93bki675x2CdKP8-h0mLD2zDgf26ptjxSNRiKFiIlM",  
 "payment-session-id": "4b7c0aec-3db3-4cd2-9b22-4d41fb964fdc",  
 "payment-status": "pending",  
 "payment-method": "creditcard",  
 "voucher-code": "LEBALLINNL10",  
 "voucher-currency": "EUR",  
 "voucher-value": 10.0  
 "voucher-status": "pending",  
 "order-price-total": 11.99,  
 "order-tax-total": 0.00,  
 "order-fee-total": 1.99,  
 "order-status": "pending",  
 "serial-number": "1234",  
 "confirmation-date": "1611313953175"  
 }  
 }  
 }

#### Provider Text

See [GetProvicerTexts] method in the [Mimas integration] documentation for the source of these fields.

##### Provider Text request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

GET /api/rest/v1/provider-texts/ HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

GET /api/rest/v1/provider-texts/:provider-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Provider Text response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

{  
 "data": [  
 {  
 "id": "f74bb480-d6df-47ad-8a17-8d7b1b176661",  
 "type": "voucher-text",  
 "attributes": {  
 "provider-code": "ABOUTYOU",  
 "kind": "title",  
 "title": "Produktname",  
 "body": "About You-Geschenkkarte"  
 }  
 },  
 {  
 "id": "a598bf3b-abe0-4fc4-80fa-135a84510c81",  
 "type": "voucher-text",  
 "attributes": {  
 "provider-code": "ABOUTYOU",  
 "kind": "subtitle",  
 "title": "Slogan",  
 "body": "Hol Dir per E-Mail eine About You-Geschenkkarte."  
 }  
 }  
 ]  
 }

#### Content

See [Content Service] for more information on the source of these fields.

##### Content request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

GET /api/rest/v1/content/i18next.json HTTP/1.1  
 Authorization: No Auth  
 Accept: application/json

##### Content response

HTTP/1.1 200 OK  
 Content-Type: application/json

{  
 "en": {  
 "world.hello": "Hello, World!",  
 "example.text": "Lorem ipsum dolor sit amet...",  
 "button.cancel": "Cancel",  
 "radio.options": {  
 "option-1": "Selection 1",  
 "option-2": "Selection 2",  
 }  
 },  
 "de": {  
 "world.hello": "Hallo, Welt!",  
 "example.text": "Lorem ipsum dolor sit amet...",  
 "button.cancel": "Abbrechen"  
 "radio.options": {  
 "option-1": "Auswahl 1",  
 "option-2": "Auswahl 2",  
 }  
 }  
 }

#### Verification

See for more information on the source of these fields.

##### Verification request by phone number

POST /api/rest/v1//auth/phone-number-verification HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": {  
 "id": "38f4dfef-9baa-4302-831f-e41df4ce4084",  
 "type": "phone-number-verification",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "sms-confirmation-code": "716569"  
 }  
 }  
 }

##### Verification by phone number response

HTTP/1.1 200 OK  
 Content-Type: application/json

{  
 "data": {  
 "type": "phone-number-verification",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "ok": false,  
 "remaining-number-of-attempts": 4,  
 "remaining-expiration-time": 0  
 }  
 }  
 }

##### Verification request by sms

POST /api/rest/v1/auth/sms-verification-code HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": {  
 "id": "48f4dfef-9baa-4302-831f-e41df4ce4084",  
 "type": "sms-verification-code",  
 "attributes": {  
 "phone-number": "+381691994655"  
 }  
 }  
 }

##### Verification by sms number response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "type": "sms-verification-code",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "ok": true,  
 "remaining-number-of-attempts": 2,  
 "remaining-delay-time": 20,  
 "remaining-expiration-time": 1800  
 }  
 }  
 }

#### Cookies

##### Cookies request

POST /api/rest/v1/cookies HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": [  
 {  
 "type": "cookie",  
 "id": "shopId",  
 "attributes": {  
 "name": "shopId",  
 "value": "1234"  
 }  
 },  
 {  
 "type": "cookie",  
 "id": "ns\_dev\_id",  
 "attributes": {  
 "name": "ns\_dev\_id",  
 "value": "device\_id"  
 }  
 }  
 ]  
 }

##### Cookies response

HTTP/1.1 200 OK

#### Documentation

Returns list of Json Schemas for request and response for requested rest endpoint. Use exact endpoint plus schema url parameter. For example, /api/rest/v1/vouchers?schema will return json schemas for /api/rest/v1/vouchers endpoint.

##### Documentation request

GET /api/rest/v1/vouchers?schema HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Documentation response

Complete Response is omitted because of size , here is just a part :

HTTP/1.1 200 OK

{  
 "required": [  
 "createdDate",  
 "notificationStatus",  
 "notificationTarget",  
 "notificationType",  
 "orderFeeTotal",  
 "orderPriceTotal",  
 "orderStatus",  
 "orderTaxTotal",  
 "paymentMethod",  
 "paymentProvider",  
 "paymentStatus",  
 "shortId",  
 "voucherCode",  
 "voucherCurrency",  
 "voucherStatus",  
 "voucherValue"  
 ]  
 }

#### i18 internationalization

* Retrieves i18next json based on requested language

##### i18 request

GET api/rest/v1/content/i18next.json/de HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

## Edge Service

The Edge Service provides the interface between the public REST API and internal AMQP API.

### TODO

#### TODO version 1.0

* [x] Test environment and credentials
* [x] specify voucher content fields
* [x] specify voucher pricing fields
* [x] specify payment method fields

#### TODO version 1.0-next

* [ ] refactor entities and apply relationships
* [ ] provide full resource schemas
* [ ] enforce validation of resource schemas

### Test environment and credentials

TODO

| Username | Password | Description |
| --- | --- | --- |
| [test@test.org](mailto:test@test.org) | test | just protected |
| [test@gmail.com](mailto:test@gmail.com) | test | just testing |
| testtest | test | qa |

### Edge Service Registry

The Edge Service Registry is responsible to maintain an index of all currently available Edge Services and configure all its known [Load-Balancers] accordingly.

As of writing no hosting solution has been chosen yet. Therefor the actual API which is used to configure the load-balancers remains unspecified for now.

All Edge Service Instances **must** register themselves on the registry on startup and un-register on shutdown.

### Edge Service Relay

Every Edge Service will be provided with a local relay connecting its instances to the central Message Broker cluster. The message exchange routing of the relay is configured in such a way, that only messages relevant to its connected Edge Services is routed to it from the broker cluster.

See [Message Relay] for more information.

### Edge Service Client

Each Service that is to be served by the Edge Service is represented by a so-called Edge Service Client. It is responsible to provide the implementation of the public REST API for that given service.

### ReST API

The Edge Service API acts as translation layer between the Services and the Frontend. The actual logic is implemented in separate [Edge Service Client]s for each individual Service consumed by the API or API endpoint provided by the API, respectively.

All public-facing endpoints **must** adhere to the [JSONAPI] specification and be subject to strict [Access Control].

#### Login

##### Login request

| Variables | Description | Example |
| --- | --- | --- |
| ICP\_SHOP\_ID | see [Test environment and credentials] |  |
| ICP\_SHOP\_PASSPHRASE | see [Test environment and credentials] |  |
| ICP\_SHOP\_DIGEST | Derived from ${ICP\_SHOP\_ID} and ${ICP\_SHOP\_PASSPHRASE} | MzAwMDAwMTY6dG9wLXNlY3JldC1wYXNzcGhyYXNl |

POST /api/rest/v1/auth/login HTTP/1.1  
 Authorization: Basic ${ICP\_SHOP\_DIGEST}  
 Accept: application/vnd.api+json

##### Login response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

##### Cookies

| Name | Value | Domain | Path | Expires | HttpOnly | Secure |
| --- | --- | --- | --- | --- | --- | --- |
| SESSION | MmQ5ZjkxOGItMTM4My00M2E0LWI1MDYtYmUwZjYwMzJiZDY5 | master.env.api.shop1.rocks | / | Seassion | true | false |

#### Logout

##### Logout request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

POST /api/rest/v1/auth/logout HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Logout response

HTTP/1.1 204 No Content

#### Voucher list

See [VoucherWebList] method in the [Mimas integration] documentation for the source of these fields.

##### Voucher list request

GET /api/rest/v1/vouchers HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher list response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data":[  
 {  
 "id":"f634f09f-9a06-4297-a326-0d703938b702",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL10",  
 "currency":"EUR",  
 "ean-code":"8717472242211",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 10 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 10",  
 "value":10.0,  
 "price-total":11.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods": {  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 }  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 }  
 }  
 },  
 {  
 "id":"a3473e20-19b4-44fe-8389-67a511854759",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL15",  
 "currency":"EUR",  
 "ean-code":"8717472242129",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 15 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 15",  
 "value":15.0,  
 "price-total":16.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods":[  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 }  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 ]  
 }  
 }  
 ]  
 }

#### Voucher detail

##### Voucher detail request

GET /api/rest/v1/vouchers/:voucher-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher detail response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data":{  
 "id":"f634f09f-9a06-4297-a326-0d703938b702",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL10",  
 "currency":"EUR",  
 "ean-code":"8717472242211",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 10 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 10",  
 "value":10.0,  
 "price-total":11.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods": {  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 },  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 }  
 }  
 }  
 }

#### Voucher order

See [VoucherWebReserve] method in the [Mimas integration] documentation for the source of these fields.

##### Voucher order request

| Variables | Description | Example | | ------------------------| -------------------------------------------- | --------------------------------------------------------------- | |

POST /api/rest/v1/voucher-orders HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "voucher-code": "VODADE15"  
 "payment-method": "Wirecard VISA",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "provider-text-url": "https://provider.texts",  
   
   
 "order-success-redirect-url": "/success",  
 "order-failed-redirect-url": "failed",  
 "order-canceled-redirect-url": "/cancel"  
   
 }  
 }  
 }

| field name | type | constrains | description | example |
| --- | --- | --- | --- | --- |
| id | UUID | globally unique | globally unique UUID | c91feaf3-5968-4b60-aa5b-62fe7fdd905c |
| type | string | n/a | type of request, must be "voucher-order", "voucher-order" |  |
| voucher-code | string | n/a | code of the voucher gathered from voucher list | "VODADE15" |
| payment-method | string | n/a | on of the allowed methods for the voucher, denoted in voucher list response | "Wirecard VISA" |
| notification-type | string | currently only "email" is supported | type of the notification target | "email" |
| notification-target | string | n/a | target adress | "[me@my-email.com](mailto:me@my-email.com)" |
| order-success-redirect-url | string | valid URL | Where the wirecard should redirect if payment was successfull | protect |
| order-failed-redirect-url | string | valid URL | Where the wirecard should redirect if payment fails | protect |
| order-canceled-redirect-url | string | valid URL | Where the wirecard should redirect if payment was canceled | protect |

##### Voucher order response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "notification-status": "pending",  
 "confirmation-status": "pending",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "payment-provider": "wirecard",  
 "payment-redirect-url": "https://wpp-test.wirecard.com/?wPaymentToken=y93bki675x2CdKP8-h0mLD2zDgf26ptjxSNRiKFiIlM",  
 "payment-session-id": "4b7c0aec-3db3-4cd2-9b22-4d41fb964fdc",  
 "payment-status": "pending",  
 "payment-method": "creditcard",  
 "voucher-code": "LEBALLINNL10",  
 "voucher-currency": "EUR",  
 "voucher-value": 10.0  
 "voucher-status": "pending",  
 "order-price-total": 11.99,  
 "order-tax-total": 0.00,  
 "order-fee-total": 0.99,  
 "order-status": "pending",  
 "serial-number": "1234",  
 "confirmation-date": "0",  
 "batch-number": "1234"  
 }  
 }  
 }

#### Voucher order detail

##### Voucher order detail request

GET /api/rest/v1/voucher-orders/:order-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher order detail response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "notification-status": "pending",  
 "confirmation-status": "pending",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "payment-provider": "wirecard",  
 "payment-redirect-url": "https://wpp-test.wirecard.com/?wPaymentToken=y93bki675x2CdKP8-h0mLD2zDgf26ptjxSNRiKFiIlM",  
 "payment-session-id": "4b7c0aec-3db3-4cd2-9b22-4d41fb964fdc",  
 "payment-status": "pending",  
 "payment-method": "creditcard",  
 "voucher-code": "LEBALLINNL10",  
 "voucher-currency": "EUR",  
 "voucher-value": 10.0  
 "voucher-status": "pending",  
 "order-price-total": 11.99,  
 "order-tax-total": 0.00,  
 "order-fee-total": 1.99,  
 "order-status": "pending",  
 "serial-number": "1234",  
 "confirmation-date": "1611313953175"  
 }  
 }  
 }

#### Provider Text

See [GetProvicerTexts] method in the [Mimas integration] documentation for the source of these fields.

##### Provider Text request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

GET /api/rest/v1/provider-texts/ HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

GET /api/rest/v1/provider-texts/:provider-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Provider Text response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

{  
 "data": [  
 {  
 "id": "f74bb480-d6df-47ad-8a17-8d7b1b176661",  
 "type": "voucher-text",  
 "attributes": {  
 "provider-code": "ABOUTYOU",  
 "kind": "title",  
 "title": "Produktname",  
 "body": "About You-Geschenkkarte"  
 }  
 },  
 {  
 "id": "a598bf3b-abe0-4fc4-80fa-135a84510c81",  
 "type": "voucher-text",  
 "attributes": {  
 "provider-code": "ABOUTYOU",  
 "kind": "subtitle",  
 "title": "Slogan",  
 "body": "Hol Dir per E-Mail eine About You-Geschenkkarte."  
 }  
 }  
 ]  
 }

#### Content

See [Content Service] for more information on the source of these fields.

##### Content request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

GET /api/rest/v1/content/i18next.json HTTP/1.1  
 Authorization: No Auth  
 Accept: application/json

##### Content response

HTTP/1.1 200 OK  
 Content-Type: application/json

{  
 "en": {  
 "world.hello": "Hello, World!",  
 "example.text": "Lorem ipsum dolor sit amet...",  
 "button.cancel": "Cancel",  
 "radio.options": {  
 "option-1": "Selection 1",  
 "option-2": "Selection 2",  
 }  
 },  
 "de": {  
 "world.hello": "Hallo, Welt!",  
 "example.text": "Lorem ipsum dolor sit amet...",  
 "button.cancel": "Abbrechen"  
 "radio.options": {  
 "option-1": "Auswahl 1",  
 "option-2": "Auswahl 2",  
 }  
 }  
 }

#### Verification

See for more information on the source of these fields.

##### Verification request by phone number

POST /api/rest/v1//auth/phone-number-verification HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": {  
 "id": "38f4dfef-9baa-4302-831f-e41df4ce4084",  
 "type": "phone-number-verification",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "sms-confirmation-code": "716569"  
 }  
 }  
 }

##### Verification by phone number response

HTTP/1.1 200 OK  
 Content-Type: application/json

{  
 "data": {  
 "type": "phone-number-verification",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "ok": false,  
 "remaining-number-of-attempts": 4,  
 "remaining-expiration-time": 0  
 }  
 }  
 }

##### Verification request by sms

POST /api/rest/v1/auth/sms-verification-code HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": {  
 "id": "48f4dfef-9baa-4302-831f-e41df4ce4084",  
 "type": "sms-verification-code",  
 "attributes": {  
 "phone-number": "+381691994655"  
 }  
 }  
 }

##### Verification by sms number response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "type": "sms-verification-code",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "ok": true,  
 "remaining-number-of-attempts": 2,  
 "remaining-delay-time": 20,  
 "remaining-expiration-time": 1800  
 }  
 }  
 }

#### Cookies

##### Cookies request

POST /api/rest/v1/cookies HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": [  
 {  
 "type": "cookie",  
 "id": "shopId",  
 "attributes": {  
 "name": "shopId",  
 "value": "1234"  
 }  
 },  
 {  
 "type": "cookie",  
 "id": "ns\_dev\_id",  
 "attributes": {  
 "name": "ns\_dev\_id",  
 "value": "device\_id"  
 }  
 }  
 ]  
 }

##### Cookies response

HTTP/1.1 200 OK

#### Documentation

Returns list of Json Schemas for request and response for requested rest endpoint. Use exact endpoint plus schema url parameter. For example, /api/rest/v1/vouchers?schema will return json schemas for /api/rest/v1/vouchers endpoint.

##### Documentation request

GET /api/rest/v1/vouchers?schema HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Documentation response

Complete Response is omitted because of size , here is just a part :

HTTP/1.1 200 OK

{  
 "required": [  
 "createdDate",  
 "notificationStatus",  
 "notificationTarget",  
 "notificationType",  
 "orderFeeTotal",  
 "orderPriceTotal",  
 "orderStatus",  
 "orderTaxTotal",  
 "paymentMethod",  
 "paymentProvider",  
 "paymentStatus",  
 "shortId",  
 "voucherCode",  
 "voucherCurrency",  
 "voucherStatus",  
 "voucherValue"  
 ]  
 }

#### i18 internationalization

* Retrieves i18next json based on requested language

##### i18 request

GET api/rest/v1/content/i18next.json/de HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

## Edge Service

The Edge Service provides the interface between the public REST API and internal AMQP API.

### TODO

#### TODO version 1.0

* [x] Test environment and credentials
* [x] specify voucher content fields
* [x] specify voucher pricing fields
* [x] specify payment method fields

#### TODO version 1.0-next

* [ ] refactor entities and apply relationships
* [ ] provide full resource schemas
* [ ] enforce validation of resource schemas

### Test environment and credentials

TODO

| Username | Password | Description |
| --- | --- | --- |
| [test@test.org](mailto:test@test.org) | test | just protected |
| [test@gmail.com](mailto:test@gmail.com) | test | just testing |
| testtest | test | qa |

### Edge Service Registry

The Edge Service Registry is responsible to maintain an index of all currently available Edge Services and configure all its known [Load-Balancers] accordingly.

As of writing no hosting solution has been chosen yet. Therefor the actual API which is used to configure the load-balancers remains unspecified for now.

All Edge Service Instances **must** register themselves on the registry on startup and un-register on shutdown.

### Edge Service Relay

Every Edge Service will be provided with a local relay connecting its instances to the central Message Broker cluster. The message exchange routing of the relay is configured in such a way, that only messages relevant to its connected Edge Services is routed to it from the broker cluster.

See [Message Relay] for more information.

### Edge Service Client

Each Service that is to be served by the Edge Service is represented by a so-called Edge Service Client. It is responsible to provide the implementation of the public REST API for that given service.

### ReST API

The Edge Service API acts as translation layer between the Services and the Frontend. The actual logic is implemented in separate [Edge Service Client]s for each individual Service consumed by the API or API endpoint provided by the API, respectively.

All public-facing endpoints **must** adhere to the [JSONAPI] specification and be subject to strict [Access Control].

#### Login

##### Login request

| Variables | Description | Example |
| --- | --- | --- |
| ICP\_SHOP\_ID | see [Test environment and credentials] |  |
| ICP\_SHOP\_PASSPHRASE | see [Test environment and credentials] |  |
| ICP\_SHOP\_DIGEST | Derived from ${ICP\_SHOP\_ID} and ${ICP\_SHOP\_PASSPHRASE} | MzAwMDAwMTY6dG9wLXNlY3JldC1wYXNzcGhyYXNl |

POST /api/rest/v1/auth/login HTTP/1.1  
 Authorization: Basic ${ICP\_SHOP\_DIGEST}  
 Accept: application/vnd.api+json

##### Login response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

##### Cookies

| Name | Value | Domain | Path | Expires | HttpOnly | Secure |
| --- | --- | --- | --- | --- | --- | --- |
| SESSION | MmQ5ZjkxOGItMTM4My00M2E0LWI1MDYtYmUwZjYwMzJiZDY5 | master.env.api.shop1.rocks | / | Seassion | true | false |

#### Logout

##### Logout request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

POST /api/rest/v1/auth/logout HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Logout response

HTTP/1.1 204 No Content

#### Voucher list

See [VoucherWebList] method in the [Mimas integration] documentation for the source of these fields.

##### Voucher list request

GET /api/rest/v1/vouchers HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher list response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data":[  
 {  
 "id":"f634f09f-9a06-4297-a326-0d703938b702",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL10",  
 "currency":"EUR",  
 "ean-code":"8717472242211",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 10 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 10",  
 "value":10.0,  
 "price-total":11.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods": {  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 }  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 }  
 }  
 },  
 {  
 "id":"a3473e20-19b4-44fe-8389-67a511854759",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL15",  
 "currency":"EUR",  
 "ean-code":"8717472242129",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 15 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 15",  
 "value":15.0,  
 "price-total":16.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods":[  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 }  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 ]  
 }  
 }  
 ]  
 }

#### Voucher detail

##### Voucher detail request

GET /api/rest/v1/vouchers/:voucher-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher detail response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data":{  
 "id":"f634f09f-9a06-4297-a326-0d703938b702",  
 "type":"voucher",  
 "attributes":{  
 "code":"LEBALLINNL10",  
 "currency":"EUR",  
 "ean-code":"8717472242211",  
 "encryption-type":0,  
 "kind":0,  
 "name":"Lebara All-in NL 10 EUR",  
 "provider-code":"LEBARA",  
 "provider-name":"Lebara",  
 "short-name":"All-in NL 10",  
 "value":10.0,  
 "price-total":11.99,  
 "price-tax":0.00,  
 "price-fee":1.99,  
 "payment-methods": {  
 "wirecard-creditcard":{  
 "name":"Credit Card",  
 "provider-name":"Wirecard",  
 "image-url":"https://wirecard.com/logo.png"  
 },  
 "wirecard-paypal":{  
 "name":"PayPal",  
 "provider-name":"Wirecard"  
 }  
 }  
 }  
 }  
 }

#### Voucher order

See [VoucherWebReserve] method in the [Mimas integration] documentation for the source of these fields.

##### Voucher order request

| Variables | Description | Example | | ------------------------| -------------------------------------------- | --------------------------------------------------------------- | |

POST /api/rest/v1/voucher-orders HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "voucher-code": "VODADE15"  
 "payment-method": "Wirecard VISA",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "provider-text-url": "https://provider.texts",  
   
   
 "order-success-redirect-url": "/success",  
 "order-failed-redirect-url": "failed",  
 "order-canceled-redirect-url": "/cancel"  
   
 }  
 }  
 }

| field name | type | constrains | description | example |
| --- | --- | --- | --- | --- |
| id | UUID | globally unique | globally unique UUID | c91feaf3-5968-4b60-aa5b-62fe7fdd905c |
| type | string | n/a | type of request, must be "voucher-order", "voucher-order" |  |
| voucher-code | string | n/a | code of the voucher gathered from voucher list | "VODADE15" |
| payment-method | string | n/a | on of the allowed methods for the voucher, denoted in voucher list response | "Wirecard VISA" |
| notification-type | string | currently only "email" is supported | type of the notification target | "email" |
| notification-target | string | n/a | target adress | "[me@my-email.com](mailto:me@my-email.com)" |
| order-success-redirect-url | string | valid URL | Where the wirecard should redirect if payment was successfull | protect |
| order-failed-redirect-url | string | valid URL | Where the wirecard should redirect if payment fails | protect |
| order-canceled-redirect-url | string | valid URL | Where the wirecard should redirect if payment was canceled | protect |

##### Voucher order response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "notification-status": "pending",  
 "confirmation-status": "pending",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "payment-provider": "wirecard",  
 "payment-redirect-url": "https://wpp-test.wirecard.com/?wPaymentToken=y93bki675x2CdKP8-h0mLD2zDgf26ptjxSNRiKFiIlM",  
 "payment-session-id": "4b7c0aec-3db3-4cd2-9b22-4d41fb964fdc",  
 "payment-status": "pending",  
 "payment-method": "creditcard",  
 "voucher-code": "LEBALLINNL10",  
 "voucher-currency": "EUR",  
 "voucher-value": 10.0  
 "voucher-status": "pending",  
 "order-price-total": 11.99,  
 "order-tax-total": 0.00,  
 "order-fee-total": 0.99,  
 "order-status": "pending",  
 "serial-number": "1234",  
 "confirmation-date": "0",  
 "batch-number": "1234"  
 }  
 }  
 }

#### Voucher order detail

##### Voucher order detail request

GET /api/rest/v1/voucher-orders/:order-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Voucher order detail response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "id": "c91feaf3-5968-4b60-aa5b-62fe7fdd905c",  
 "type": "voucher-order",  
 "attributes": {  
 "notification-status": "pending",  
 "confirmation-status": "pending",  
 "notification-type": "email",  
 "notification-target": "me@my-email.com",  
 "payment-provider": "wirecard",  
 "payment-redirect-url": "https://wpp-test.wirecard.com/?wPaymentToken=y93bki675x2CdKP8-h0mLD2zDgf26ptjxSNRiKFiIlM",  
 "payment-session-id": "4b7c0aec-3db3-4cd2-9b22-4d41fb964fdc",  
 "payment-status": "pending",  
 "payment-method": "creditcard",  
 "voucher-code": "LEBALLINNL10",  
 "voucher-currency": "EUR",  
 "voucher-value": 10.0  
 "voucher-status": "pending",  
 "order-price-total": 11.99,  
 "order-tax-total": 0.00,  
 "order-fee-total": 1.99,  
 "order-status": "pending",  
 "serial-number": "1234",  
 "confirmation-date": "1611313953175"  
 }  
 }  
 }

#### Provider Text

See [GetProvicerTexts] method in the [Mimas integration] documentation for the source of these fields.

##### Provider Text request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

GET /api/rest/v1/provider-texts/ HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

GET /api/rest/v1/provider-texts/:provider-id HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Provider Text response

HTTP/1.1 201 Created  
 Content-Type: application/vnd.api+json

{  
 "data": [  
 {  
 "id": "f74bb480-d6df-47ad-8a17-8d7b1b176661",  
 "type": "voucher-text",  
 "attributes": {  
 "provider-code": "ABOUTYOU",  
 "kind": "title",  
 "title": "Produktname",  
 "body": "About You-Geschenkkarte"  
 }  
 },  
 {  
 "id": "a598bf3b-abe0-4fc4-80fa-135a84510c81",  
 "type": "voucher-text",  
 "attributes": {  
 "provider-code": "ABOUTYOU",  
 "kind": "subtitle",  
 "title": "Slogan",  
 "body": "Hol Dir per E-Mail eine About You-Geschenkkarte."  
 }  
 }  
 ]  
 }

#### Content

See [Content Service] for more information on the source of these fields.

##### Content request

| Variables | Description | Example |
| --- | --- | --- |
|  |  |  |

GET /api/rest/v1/content/i18next.json HTTP/1.1  
 Authorization: No Auth  
 Accept: application/json

##### Content response

HTTP/1.1 200 OK  
 Content-Type: application/json

{  
 "en": {  
 "world.hello": "Hello, World!",  
 "example.text": "Lorem ipsum dolor sit amet...",  
 "button.cancel": "Cancel",  
 "radio.options": {  
 "option-1": "Selection 1",  
 "option-2": "Selection 2",  
 }  
 },  
 "de": {  
 "world.hello": "Hallo, Welt!",  
 "example.text": "Lorem ipsum dolor sit amet...",  
 "button.cancel": "Abbrechen"  
 "radio.options": {  
 "option-1": "Auswahl 1",  
 "option-2": "Auswahl 2",  
 }  
 }  
 }

#### Verification

See for more information on the source of these fields.

##### Verification request by phone number

POST /api/rest/v1//auth/phone-number-verification HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": {  
 "id": "38f4dfef-9baa-4302-831f-e41df4ce4084",  
 "type": "phone-number-verification",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "sms-confirmation-code": "716569"  
 }  
 }  
 }

##### Verification by phone number response

HTTP/1.1 200 OK  
 Content-Type: application/json

{  
 "data": {  
 "type": "phone-number-verification",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "ok": false,  
 "remaining-number-of-attempts": 4,  
 "remaining-expiration-time": 0  
 }  
 }  
 }

##### Verification request by sms

POST /api/rest/v1/auth/sms-verification-code HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": {  
 "id": "48f4dfef-9baa-4302-831f-e41df4ce4084",  
 "type": "sms-verification-code",  
 "attributes": {  
 "phone-number": "+381691994655"  
 }  
 }  
 }

##### Verification by sms number response

HTTP/1.1 200 OK  
 Content-Type: application/vnd.api+json

{  
 "data": {  
 "type": "sms-verification-code",  
 "attributes": {  
 "phone-number": "+381691994655",  
 "ok": true,  
 "remaining-number-of-attempts": 2,  
 "remaining-delay-time": 20,  
 "remaining-expiration-time": 1800  
 }  
 }  
 }

#### Cookies

##### Cookies request

POST /api/rest/v1/cookies HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json  
   
 {  
 "data": [  
 {  
 "type": "cookie",  
 "id": "shopId",  
 "attributes": {  
 "name": "shopId",  
 "value": "1234"  
 }  
 },  
 {  
 "type": "cookie",  
 "id": "ns\_dev\_id",  
 "attributes": {  
 "name": "ns\_dev\_id",  
 "value": "device\_id"  
 }  
 }  
 ]  
 }

##### Cookies response

HTTP/1.1 200 OK

#### Documentation

Returns list of Json Schemas for request and response for requested rest endpoint. Use exact endpoint plus schema url parameter. For example, /api/rest/v1/vouchers?schema will return json schemas for /api/rest/v1/vouchers endpoint.

##### Documentation request

GET /api/rest/v1/vouchers?schema HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json

##### Documentation response

Complete Response is omitted because of size , here is just a part :

HTTP/1.1 200 OK

{  
 "required": [  
 "createdDate",  
 "notificationStatus",  
 "notificationTarget",  
 "notificationType",  
 "orderFeeTotal",  
 "orderPriceTotal",  
 "orderStatus",  
 "orderTaxTotal",  
 "paymentMethod",  
 "paymentProvider",  
 "paymentStatus",  
 "shortId",  
 "voucherCode",  
 "voucherCurrency",  
 "voucherStatus",  
 "voucherValue"  
 ]  
 }

#### i18 internationalization

* Retrieves i18next json based on requested language

##### i18 request

GET api/rest/v1/content/i18next.json/de HTTP/1.1  
 Authorization: No Auth  
 Accept: application/vnd.api+json



Figure : C:\puzzles-cloud-docs-site\images\logo.png