Template version: 1.0

[[doc-title]]

[[doc-subtitle]]

[[doc-type]]

[[doc-company]]
[[doc-company-url]]

Document Number: [[doc-id]]
Version: [[doc-version]]
Release Date: [[release-date]]

Document History

Author Name: [[doc-author]]
Author Contact: [[doc-author-email]]

Document Revision

Version Number	Release Date	Status	Change Description	Author
[[doc-version]]	[[doc-release-date]]	Draft	[[change-description]]	[[doc-author-name]]

Intellectual Property Rights

This document is exclusive property of [[doc-company]] and all rights are reserved.

Table of Contents

Introduction 12

References 12

Markdown3 14

Edge Service 14

TODO 14

TODO version 1.0 14

TODO version 1.0-next 14

Test environment and credentials 14

Edge Service Registry 14

Edge Service Relay 14

Edge Service Client 14

ReST API 14

Login 15

Login request 15

Login response 15

Cookies 15

Logout 15

Logout request 15

Logout response 15

Voucher list 15

Voucher list request 15

Voucher list response 15

Voucher detail 15

Voucher detail request 15

Voucher detail response 15

Voucher order 15

Voucher order request 15

Voucher order response 16

Voucher order detail 16

Voucher order detail request 16

Voucher order detail response 16

Provider Text 16

Provider Text request 16

Provider Text response 17

Content 17

Content request 17

Content response 17

Verification 17

Verification request by phone number 17

Verification by phone number response 17

Verification request by sms 17

Verification by sms number response 17

Cookies 17

Cookies request 17

Cookies response 17

Documentation 17
Documentation request 17
Documentation response 17
i18 internationalization 17
i18 request 17

Markdown3 18

Edge Service 18
TODO 18
TODO version 1.0 18
TODO version 1.0-next 18
Test environment and credentials 18
Edge Service Registry 18
Edge Service Relay 18
Edge Service Client 18
ReST API 18
Login 19
Login request 19
Login response 19
Cookies 19
Logout 19
Logout request 19
Logout response 19
Voucher list 19
Voucher list request 19
Voucher list response 19
Voucher detail 19
Voucher detail request 19
Voucher detail response 19
Voucher order 19
Voucher order request 19
Voucher order response 20
Voucher order detail 20
Voucher order detail request 20
Voucher order detail response 20
Provider Text 20
Provider Text request 20
Provider Text response 21
Content 21
Content request 21
Content response 21
Verification 21
Verification request by phone number 21
Verification by phone number response 21
Verification request by sms 21
Verification by sms number response 21
Cookies 21
Cookies request 21
Cookies response 21
Documentation 21

Documentation request 21
Documentation response 21
i18 internationalization 21
i18 request 21

Markdown4-2 22

From today's featured article 22
Did you know ... 22
In the news 22
On this day 22
March 2 22
Today's featured picture 23

Armed conflicts and attacks 23
Health and environment 23

Markdown3 24

Edge Service 24
TODO 24
TODO version 1.0 24
TODO version 1.0-next 24
Test environment and credentials 24
Edge Service Registry 24
Edge Service Relay 24
Edge Service Client 24
ReST API 24
Login 25
Login request 25
Login response 25
Cookies 25
Logout 25
Logout request 25
Logout response 25
Voucher list 25
Voucher list request 25
Voucher list response 25
Voucher detail 25
Voucher detail request 25
Voucher detail response 25
Voucher order 25
Voucher order request 25
Voucher order response 26
Voucher order detail 26
Voucher order detail request 26
Voucher order detail response 26
Provider Text 26
Provider Text request 26
Provider Text response 27
Content 27
Content request 27

Content response 27
Verification 27
Verification request by phone number 27
Verification by phone number response 27
Verification request by sms 27
Verification by sms number response 27
Cookies 27
Cookies request 27
Cookies response 27
Documentation 27
Documentation request 27
Documentation response 27
i18 internationalization 27
i18 request 27
Markdown3 28
Edge Service 28
TODO 28
TODO version 1.0 28
TODO version 1.0-next 28
Test environment and credentials 28
Edge Service Registry 28
Edge Service Relay 28
Edge Service Client 28
ReST API 28
Login 29
Login request 29
Login response 29
Cookies 29
Logout 29
Logout request 29
Logout response 29
Voucher list 29
Voucher list request 29
Voucher list response 29
Voucher detail 29
Voucher detail request 29
Voucher detail response 29
Voucher order 29
Voucher order request 29
Voucher order response 30
Voucher order detail 30
Voucher order detail request 30
Voucher order detail response 30
Provider Text 30
Provider Text request 30
Provider Text response 31
Content 31
Content request 31
Content response 31

Verification 31
Verification request by phone number 31
Verification by phone number response 31
Verification request by sms 31
Verification by sms number response 31
Cookies 31
Cookies request 31
Cookies response 31
Documentation 31
Documentation request 31
Documentation response 31
i18 internationalization 31
i18 request 31

Markdown4-2 32

From today's featured article 32
Did you know ... 32
In the news 32
On this day 32
March 2 32
Today's featured picture 33

Armed conflicts and attacks 33
Health and environment 33

Markdown3 34

Edge Service 34
TODO 34
TODO version 1.0 34
TODO version 1.0-next 34
Test environment and credentials 34
Edge Service Registry 34
Edge Service Relay 34
Edge Service Client 34
ReST API 34
Login 35
Login request 35
Login response 35
Cookies 35
Logout 35
Logout request 35
Logout response 35
Voucher list 35
Voucher list request 35
Voucher list response 35
Voucher detail 35
Voucher detail request 35
Voucher detail response 35
Voucher order 35
Voucher order request 35

Voucher order response 36
Voucher order detail 36
Voucher order detail request 36
Voucher order detail response 36
Provider Text 36
Provider Text request 36
Provider Text response 37
Content 37
Content request 37
Content response 37
Verification 37
Verification request by phone number 37
Verification by phone number response 37
Verification request by sms 37
Verification by sms number response 37
Cookies 37
Cookies request 37
Cookies response 37
Documentation 37
Documentation request 37
Documentation response 37
i18 internationalization 37
i18 request 37

UC3 Just testing 38

Edge Service 38
TODO 38
TODO version 1.0 38
TODO version 1.0-next 38
Test environment and credentials 38
Edge Service Registry 38
Edge Service Relay 38
Edge Service Client 38
ReST API 38
Login 39
Login request 39
Login response 39
Cookies 39
Logout 39
Logout request 39
Logout response 39
Voucher list 39
Voucher list request 39
Voucher list response 39
Voucher detail 39
Voucher detail request 39
Voucher detail response 39
Voucher order 39
Voucher order request 39
Voucher order response 40

Voucher order detail 40
Voucher order detail request 40
Voucher order detail response 40
Provider Text 40
Provider Text request 40
Provider Text response 41
Content 41
Content request 41
Content response 41
Verification 41
Verification request by phone number 41
Verification by phone number response 41
Verification request by sms 41
Verification by sms number response 41
Cookies 41
Cookies request 41
Cookies response 41
Documentation 41
Documentation request 41
Documentation response 41
i18 internationalization 41
i18 request 41

UC3 Just testing 42

Edge Service 42
TODO 42
TODO version 1.0 42
TODO version 1.0-next 42
Test environment and credentials 42
Edge Service Registry 42
Edge Service Relay 42
Edge Service Client 42
ReST API 42
Login 43
Login request 43
Login response 43
Cookies 43
Logout 43
Logout request 43
Logout response 43
Voucher list 43
Voucher list request 43
Voucher list response 43
Voucher detail 43
Voucher detail request 43
Voucher detail response 43
Voucher order 43
Voucher order request 43
Voucher order response 44
Voucher order detail 44

Voucher order detail request 44
Voucher order detail response 44
Provider Text 44
Provider Text request 44
Provider Text response 45
Content 45
Content request 45
Content response 45
Verification 45
Verification request by phone number 45
Verification by phone number response 45
Verification request by sms 45
Verification by sms number response 45
Cookies 45
Cookies request 45
Cookies response 45
Documentation 45
Documentation request 45
Documentation response 45
i18 internationalization 45
i18 request 45

Backticks 46

Backticks MD file 46
Backtickts in heading 46

End of file 46

Backticks 47

Backticks MD file 47
Backtickts in heading 47

End of file 47

Backticks 48

Backticks MD file 48
Backtickts in heading 48

End of file 48

Just checking tables - heading 1 49

Tables are under heading 2 49

test2 50

Markdown2-2markdown2-2markdown2-2markdown2-2markdown2-2markdown2-2markdown2-2
Formatted Text Examples 50
Font buttons: 50
Text Format: 50
Text Justification: 50
Useful markdown links: 50
I just have to store some code here, if you don't mind 51
I'll put it here 51

And also here 51
You know the process... 51
Markdown2-2markdown2-2markdown2-2markdown2-2markdown2-2markdown2-2markdown2-2
Formatted Text Examples 51
Font buttons: 51
Text Format: 51
Text Justification: 51
Useful markdown links: 51
I just have to store some code here, if you don't mind 52
I'll put it here 52
And also here 52
You know the process... 52

Table of Tables

Table 1: Document references 12
Table 49
Table 49

1 Introduction

1.1 References

Referenced documents in creation of this document are listed in the Table 1.

Document Reference	Document Title	Version	Filename
[1]	[[reference-doc-title]]		[[reference-doc-filename]]
[2]			
[3]			
[4]			
[5]			
[6]			
[7]			
[8]			

Table 1: Document references

2 Markdown3

2.1 Edge Service

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

2.1.1 TODO

2.1.1.1 TODO version 1.0

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

2.1.1.2 TODO version 1.0-next

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

2.1.2 Test environment and credentials

TODO

Username	Password	Description
test@test.org	test	just protected
test@gmail.com	test	just testing
testtest	test	qa

2.1.3 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.

2.1.4 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.***

2.1.5 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.

2.1.6 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].

2.1.6.1 Login

2.1.6.1.1 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
```

2.1.6.1.2 Login response

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

2.1.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxOGItMTA4Mi00M2E3LWJ1MDktYmUwZjAxMWJiZDY5				true	false

2.1.6.2 Logout

2.1.6.2.1 Logout request

Variables	Description	Example

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

2.1.6.2.2 Logout response

```
HTTP/1.1 204 No Content
```

2.1.6.3 Voucher list

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

2.1.6.3.1 Voucher list request

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

2.1.6.3.2 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" } ] } } ] }
```

2.1.6.4 Voucher detail

2.1.6.4.1 Voucher detail request

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

2.1.6.4.2 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" } } } } }
```

2.1.6.5 Voucher order

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

2.1.6.5.1 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**

2.1.6.5.2 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" } } }

2.1.6 Voucher order detail

2.1.6.6.1 Voucher order detail request

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

2.1.6.6.2 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" } } }

2.1.6.7 Provider Text

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

2.1.6.7.1 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
```

2.1.6.7.2 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." } } ] }
```

2.1.6.8 Content

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

2.1.6.8.1 Content request

Variables	Description	Example

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

2.1.6.8.2 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", } } }
```

2.1.6.9 Verification

See for more information on the source of these fields.

2.1.6.9.1 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" } } }
```

2.1.6.9.2 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 } } }
```

2.1.6.9.3 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" } } }
```

2.1.6.9.4 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 } } }
```

2.1.6.10 Cookies

2.1.6.10.1 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" } } ] }
```

2.1.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

2.1.6.11 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.

2.1.6.11.1 Documentation request

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

2.1.6.11.2 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" ] }
```

2.1.6.12 i18 internationalization

 # Retrieves i18next json based on requested language

2.1.6.12.1 i18 request

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

3 Markdown3

3.1 Edge Service

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

3.1.1 TODO

3.1.1.1 TODO version 1.0

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

3.1.1.2 TODO version 1.0-next

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

3.1.2 Test environment and credentials

TODO

Username	Password	Description
test@test.org	test	just protected
test@gmail.com	test	just testing
testtest	test	qa

3.1.3 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.

3.1.4 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.***

3.1.5 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.

3.1.6 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].

3.1.6.1 Login

3.1.6.1.1 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
```

3.1.6.1.2 Login response

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

3.1.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxOGItMTA4Mi00M2E3LWJ1MDktYmUwZjAxMWJiZDY5				true	false

3.1.6.2 Logout

3.1.6.2.1 Logout request

Variables	Description	Example

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

3.1.6.2.2 Logout response

```
HTTP/1.1 204 No Content
```

3.1.6.3 Voucher list

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

3.1.6.3.1 Voucher list request

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

3.1.6.3.2 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" } ] } } ] }
```

3.1.6.4 Voucher detail

3.1.6.4.1 Voucher detail request

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

3.1.6.4.2 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" } } } } }
```

3.1.6.5 Voucher order

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

3.1.6.5.1 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"
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

3.1.6.5.2 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" } } }

3.1.6 Voucher order detail

3.1.6.6.1 Voucher order detail request

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

3.1.6.6.2 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" } } }

3.1.6.7 Provider Text

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

3.1.6.7.1 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
```

3.1.6.7.2 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." } } ] }
```

3.1.6.8 Content

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

3.1.6.8.1 Content request

Variables	Description	Example

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

3.1.6.8.2 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", } } }
```

3.1.6.9 Verification

See for more information on the source of these fields.

3.1.6.9.1 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" } } }
```

3.1.6.9.2 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 } } }
```

3.1.6.9.3 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" } } }
```

3.1.6.9.4 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 } } }
```

3.1.6.10 Cookies

3.1.6.10.1 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" } } ] }
```

3.1.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

3.1.6.11 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.

3.1.6.11.1 Documentation request

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

3.1.6.11.2 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" ] }
```

3.1.6.12 i18 internationalization

 # Retrieves i18next json based on requested language

3.1.6.12.1 i18 request

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

4 Markdown4-2

4.1 From today's featured article

Lionel Matthews (15 August 1912 -- 2 March 1944) was an Australian Army officer during World War II who was posthumously awarded the George Cross, the highest non-combat award for heroism for members of the Australian armed forces at the time. Matthews served in the 27th Brigade during the Malayan campaign, and at the surrender of Singapore in 1942 became a prisoner of war (POW). He was awarded the Military Cross for courage, energy and ability under fire during this fighting.

4.1.1 Did you know ...

that in 2017 you could finally have breakfast at Tiffany's (pictured)?
that the British mycologist Denis Garrett, "one of the last 'string and sealing wax' scientists", once bought plastic lavatory cisterns for his laboratory to use in experiments?
that sand in the stomach of the ocean surgeon helps to grind up its food?

4.1.1.1 In the news

1. More than 300 girls are kidnapped from a secondary school by armed bandits in Zamfara, Nigeria.
2. In tennis, Naomi Osaka wins the women's singles and Novak Djokovic wins the men's singles at the Australian Open (both winners pictured).

4.1.1.1.1 On this day

4.1.1.1.1.1 March 2

1444 -- *The League of Lezhë, an alliance of regional chieftains, was established in Venetian Albania with Skanderbeg as its commander.*

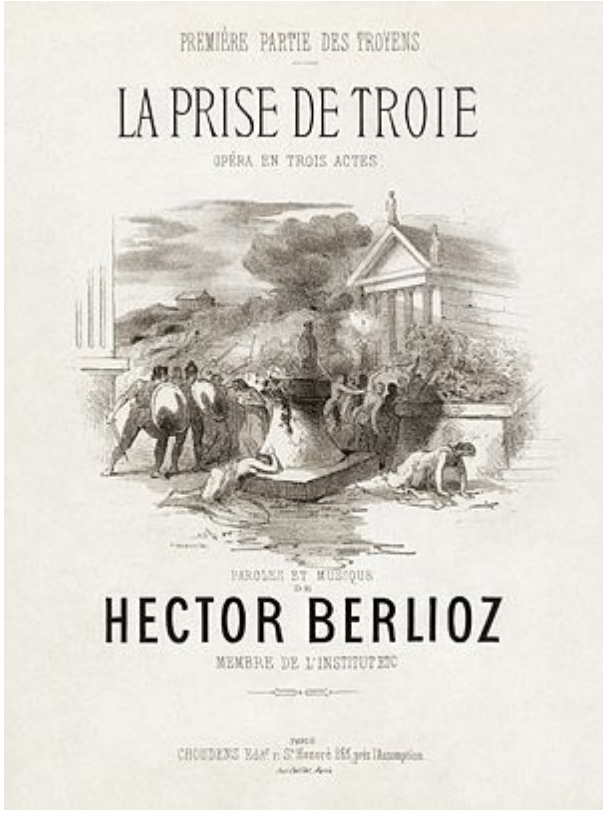
1865 -- *New Zealand Wars: Carl Sylvius Völkner (pictured), a Protestant missionary, was killed by Hauhau militants in #p#tiki for working as an agent for George Grey, Governor-General of New Zealand.*

1937 -- *The Steel Workers Organizing Committee, precursor to the United Steel Workers of America, had a major success when it signed a collective-bargaining agreement with U.S. Steel.*

1962 -- *Led by General Ne Win, the Burmese military seized power in a coup d'état.*

~~aaaaaaaaaaaaaaaaaa~~

4.1.1.1.1.1.1 Today's featured picture



4.2 Armed conflicts and attacks

Zamfara kidnapping
    1. *All 279 abducted schoolgirls from a boarding school in Zamfara State, Nigeria, are released by their armed kidnappers. (Reuters)*

4.2.1 Health and environment

COVID-19 pandemic
    1. *COVID-19 pandemic in Asia*
        # *COVID-19 pandemic in Indonesia*
            # *Indonesia reports their first two cases of Lineage B.1.1.7 variant first detected in the United Kingdom. (Liputan6.com)*
        # *COVID-19 pandemic in Iraq*
            # *Iraq receives 50,000 doses of Sinopharm vaccines as the country officially begins its vaccination campaign. (Asharq Al-Awsat)*
        # *COVID-19 pandemic in Mongolia*
            # *Mongolia surpasses 3,000 cases of COVID-19. (Montsame)*
        # *COVID-19 pandemic in the Philippines*
            # *The Philippines reports their first six cases of 501.V2 variant first detected in South Africa. (GMA News)*

5 Markdown3

5.1 Edge Service

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

5.1.1 TODO

5.1.1.1 TODO version 1.0

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

5.1.1.2 TODO version 1.0-next

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

5.1.2 Test environment and credentials

TODO

Username	Password	Description
test@test.org	test	just protected
test@gmail.com	test	just testing
testtest	test	qa

5.1.3 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.

5.1.4 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.***

5.1.5 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.

5.1.6 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].

5.1.6.1 Login

5.1.6.1.1 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
```

5.1.6.1.2 Login response

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

5.1.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxOGItMTA4Mi00M2E3LWJl[illegible]ZDY5	[illegible]	[illegible]	[illegible]	true	false

5.1.6.2 Logout

5.1.6.2.1 Logout request

Variables	Description	Example

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

5.1.6.2.2 Logout response

```
HTTP/1.1 204 No Content
```

5.1.6.3 Voucher list

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

5.1.6.3.1 Voucher list request

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

5.1.6.3.2 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" } ] } } ] }
```

5.1.6.4 Voucher detail

5.1.6.4.1 Voucher detail request

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

5.1.6.4.2 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" } } } } }
```

5.1.6.5 Voucher order

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

5.1.6.5.1 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**

5.1.6.5.2 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" } } }

5.1.6.6 Voucher order detail

5.1.6.6.1 Voucher order detail request

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

5.1.6.6.2 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" } } }

5.1.6.7 Provider Text

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

5.1.6.7.1 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
```

5.1.6.7.2 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." } } ] }
```

5.1.6.8 Content

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

5.1.6.8.1 Content request

Variables	Description	Example

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

5.1.6.8.2 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", } } }
```

5.1.6.9 Verification

See for more information on the source of these fields.

5.1.6.9.1 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" } } }
```

5.1.6.9.2 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 } } }
```

5.1.6.9.3 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" } } }
```

5.1.6.9.4 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 } } }
```

5.1.6.10 Cookies

5.1.6.10.1 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" } } ] }
```

5.1.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

5.1.6.11 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.

5.1.6.11.1 Documentation request

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

5.1.6.11.2 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" ] }
```

5.1.6.12 i18 internationalization

 # Retrieves i18next json based on requested language

5.1.6.12.1 i18 request

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

6 Markdown3

6.1 Edge Service

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

6.1.1 TODO

6.1.1.1 TODO version 1.0

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

6.1.1.2 TODO version 1.0-next

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

6.1.2 Test environment and credentials

TODO

Username	Password	Description
test@test.org	test	just protected
test@gmail.com	test	just testing
testtest	test	qa

6.1.3 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.

6.1.4 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.***

6.1.5 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.

6.1.6 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].

6.1.6.1 Login

6.1.6.1.1 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
```

6.1.6.1.2 Login response

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

6.1.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxOGItMTA4Mi00M2E3LWJ1MDktYmUwZjAxMWJiZDY5				true	false

6.1.6.2 Logout

6.1.6.2.1 Logout request

Variables	Description	Example

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

6.1.6.2.2 Logout response

```
HTTP/1.1 204 No Content
```

6.1.6.3 Voucher list

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

6.1.6.3.1 Voucher list request

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

6.1.6.3.2 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" } ] } } ] }
```

6.1.6.4 Voucher detail

6.1.6.4.1 Voucher detail request

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

6.1.6.4.2 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" } } } } }
```

6.1.6.5 Voucher order

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

6.1.6.5.1 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"**
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**

6.1.6.5.2 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" } } }

6.1.6 Voucher order detail

6.1.6.6.1 Voucher order detail request

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

6.1.6.6.2 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" } } }

6.1.6.7 Provider Text

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

6.1.6.7.1 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
```

6.1.6.7.2 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." } } ] }
```

6.1.6.8 Content

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

6.1.6.8.1 Content request

Variables	Description	Example

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

6.1.6.8.2 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", } } }
```

6.1.6.9 Verification

See for more information on the source of these fields.

6.1.6.9.1 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" } } }
```

6.1.6.9.2 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 } } }
```

6.1.6.9.3 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" } } }
```

6.1.6.9.4 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 } } }
```

6.1.6.10 Cookies

6.1.6.10.1 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" } } ] }
```

6.1.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

6.1.6.11 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.

6.1.6.11.1 Documentation request

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

6.1.6.11.2 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" ] }
```

6.1.6.12 i18 internationalization

 # Retrieves i18next json based on requested language

6.1.6.12.1 i18 request

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

7 Markdown4-2

7.1 From today's featured article

Lionel Matthews (15 August 1912 -- 2 March 1944) was an Australian Army officer during World War II who was posthumously awarded the George Cross, the highest non-combat award for heroism for members of the Australian armed forces at the time. Matthews served in the 27th Brigade during the Malayan campaign, and at the surrender of Singapore in 1942 became a prisoner of war (POW). He was awarded the Military Cross for courage, energy and ability under fire during this fighting.

7.1.1 Did you know ...

that in 2017 you could finally have breakfast at Tiffany's (pictured)?
that the British mycologist Denis Garrett, "one of the last 'string and sealing wax' scientists", once bought plastic lavatory cisterns for his laboratory to use in experiments?
that sand in the stomach of the ocean surgeon helps to grind up its food?

7.1.1.1 In the news

3. More than 300 girls are kidnapped from a secondary school by armed bandits in Zamfara, Nigeria.
4. In tennis, Naomi Osaka wins the women's singles and Novak Djokovic wins the men's singles at the Australian Open (both winners pictured).

7.1.1.1.1 On this day

7.1.1.1.1.1 March 2

1444 -- *The League of Lezhë, an alliance of regional chieftains, was established in Venetian Albania with Skanderbeg as its commander.*

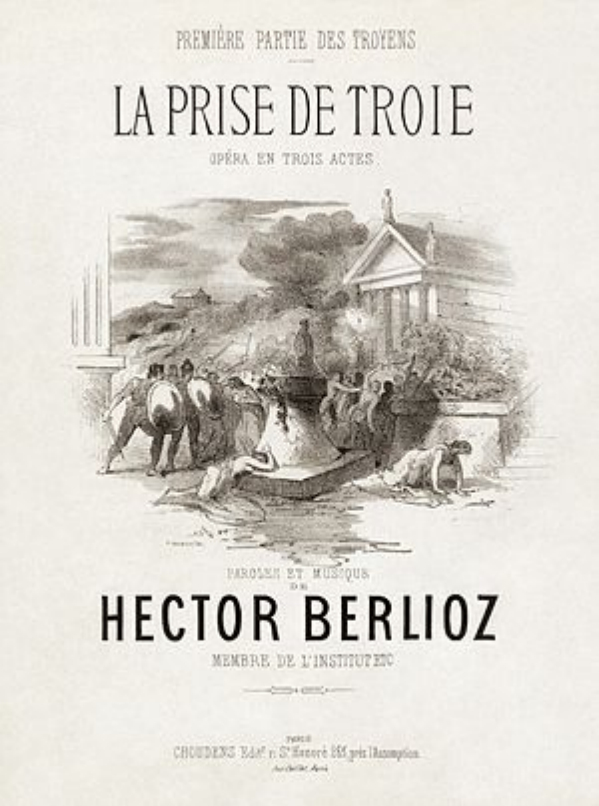
1865 -- *New Zealand Wars: Carl Sylvius Völkner (pictured), a Protestant missionary, was killed by Hauhau militants in #p#tiki for working as an agent for George Grey, Governor-General of New Zealand.*

1937 -- *The Steel Workers Organizing Committee, precursor to the United Steel Workers of America, had a major success when it signed a collective-bargaining agreement with U.S. Steel.*

1962 -- *Led by General Ne Win, the Burmese military seized power in a coup d'état.*

~~aaaaaaaaaaaaaaaaaa~~

7.1.1.1.1.1.1 Today's featured picture



7.2 Armed conflicts and attacks

Zamfara kidnapping
    1. *All 279 abducted schoolgirls from a boarding school in Zamfara State, Nigeria, are released by their armed kidnappers. (Reuters)*

7.2.1 Health and environment

COVID-19 pandemic
    1. *COVID-19 pandemic in Asia*
        # *COVID-19 pandemic in Indonesia*
            # *Indonesia reports their first two cases of Lineage B.1.1.7 variant first detected in the United Kingdom. (Liputan6.com)*
        # *COVID-19 pandemic in Iraq*
            # *Iraq receives 50,000 doses of Sinopharm vaccines as the country officially begins its vaccination campaign. (Asharq Al-Awsat)*
        # *COVID-19 pandemic in Mongolia*
            # *Mongolia surpasses 3,000 cases of COVID-19. (Montsame)*
        # *COVID-19 pandemic in the Philippines*
            # *The Philippines reports their first six cases of 501.V2 variant first detected in South Africa. (GMA News)*

8 Markdown3

8.1 Edge Service

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

8.1.1 TODO

8.1.1.1 TODO version 1.0

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

8.1.1.2 TODO version 1.0-next

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

8.1.2 Test environment and credentials

TODO

Username	Password	Description
test@test.org	test	just protected
test@gmail.com	test	just testing
testtest	test	qa

8.1.3 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.

8.1.4 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.***

8.1.5 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.

8.1.6 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].

8.1.6.1 Login

8.1.6.1.1 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
```

8.1.6.1.2 Login response

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

8.1.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxOGItMTA4Mi00M2E3LWJl[illegible]				true	false

8.1.6.2 Logout

8.1.6.2.1 Logout request

Variables	Description	Example

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

8.1.6.2.2 Logout response

```
HTTP/1.1 204 No Content
```

8.1.6.3 Voucher list

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

8.1.6.3.1 Voucher list request

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

8.1.6.3.2 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" } ] } } ] }
```

8.1.6.4 Voucher detail

8.1.6.4.1 Voucher detail request

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

8.1.6.4.2 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" } } } } }
```

8.1.6.5 Voucher order

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

8.1.6.5.1 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**

8.1.6.5.2 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" } } }
```

8.1.6 Voucher order detail

8.1.6.6.1 Voucher order detail request

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

8.1.6.6.2 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" } } }
```

8.1.6.7 Provider Text

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

8.1.6.7.1 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

8.1.6.7.2 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." } }] }

8.1.6.8 Content

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

8.1.6.8.1 Content request

Variables	Description	Example

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

8.1.6.8.2 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", } } }

8.1.6.9 Verification

See for more information on the source of these fields.

8.1.6.9.1 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" } } }

8.1.6.9.2 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 } } }

8.1.6.9.3 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" } } }

8.1.6.9.4 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 } } }

8.1.6.10 Cookies

8.1.6.10.1 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" } }] }

8.1.6.10.2 Cookies response

HTTP/1.1 200 OK

8.1.6.11 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.

8.1.6.11.1 Documentation request

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

8.1.6.11.2 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"] }

8.1.6.12 i18 internationalization

 # Retrieves i18next json based on requested language

8.1.6.12.1 i18 request

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

9 UC3 Just testing

9.1 Edge Service

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

9.1.1 TODO

9.1.1.1 TODO version 1.0

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

9.1.1.2 TODO version 1.0-next

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

9.1.2 Test environment and credentials

TODO

Username	Password	Description
test@test.org	test	just protected
test@gmail.com	test	just testing
testtest	test	qa

9.1.3 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.

9.1.4 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.***

9.1.5 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.

9.1.6 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].

9.1.6.1 Login

9.1.6.1.1 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
```

9.1.6.1.2 Login response

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

9.1.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxOGItMTA4Mi00M2E3LWJ1MDktYmUwZjAxMWJiZDY5				true	false

9.1.6.2 Logout

9.1.6.2.1 Logout request

Variables	Description	Example

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

9.1.6.2.2 Logout response

```
HTTP/1.1 204 No Content
```

9.1.6.3 Voucher list

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

9.1.6.3.1 Voucher list request

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

9.1.6.3.2 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" } ] } } ] }
```

9.1.6.4 Voucher detail

9.1.6.4.1 Voucher detail request

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

9.1.6.4.2 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" } } } } }
```

9.1.6.5 Voucher order

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

9.1.6.5.1 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"**
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**

9.1.6.5.2 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" } } }

9.1.6.6 Voucher order detail

9.1.6.6.1 Voucher order detail request

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

9.1.6.6.2 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" } } }

9.1.6.7 Provider Text

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

9.1.6.7.1 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

9.1.6.7.2 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." } }] }

9.1.6.8 Content

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

9.1.6.8.1 Content request

Variables	Description	Example

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

9.1.6.8.2 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", } } }

9.1.6.9 Verification

See for more information on the source of these fields.

9.1.6.9.1 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" } } }

9.1.6.9.2 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 } } }

9.1.6.9.3 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" } } }

9.1.6.9.4 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 } } }

9.1.6.10 Cookies

9.1.6.10.1 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" } }] }

9.1.6.10.2 Cookies response

HTTP/1.1 200 OK

9.1.6.11 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.

9.1.6.11.1 Documentation request

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

9.1.6.11.2 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"] }

9.1.6.12 i18 internationalization

 # Retrieves i18next json based on requested language

9.1.6.12.1 i18 request

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

10 UC3 Just testing

10.1 Edge Service

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

10.1.1 TODO

10.1.1.1 TODO version 1.0

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

10.1.1.2 TODO version 1.0-next

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

10.1.2 Test environment and credentials

TODO

Username	Password	Description
test@test.org	test	just protected
test@gmail.com	test	just testing
testtest	test	qa

10.1.3 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.

10.1.4 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.***

10.1.5 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.

10.1.6 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].

10.1.6.1 Login

10.1.6.1.1 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
```

10.1.6.1.2 Login response

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

10.1.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxOGItMTA4Mi00M2E3LWJl[illegible]				true	false

10.1.6.2 Logout

10.1.6.2.1 Logout request

Variables	Description	Example

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

10.1.6.2.2 Logout response

```
HTTP/1.1 204 No Content
```

10.1.6.3 Voucher list

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

10.1.6.3.1 Voucher list request

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

10.1.6.3.2 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" } ] } } ] }
```

10.1.6.4 Voucher detail

10.1.6.4.1 Voucher detail request

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

10.1.6.4.2 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" } } } } }
```

10.1.6.5 Voucher order

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

10.1.6.5.1 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"**
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**

10.1.6.5.2 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" } } }

10.1.6.6 Voucher order detail

10.1.6.6.1 Voucher order detail request

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

10.1.6.6.2 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" } } }

10.1.6.7 Provider Text

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

10.1.6.7.1 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
```

10.1.6.7.2 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." } } ] }
```

10.1.6.8 Content

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

10.1.6.8.1 Content request

Variables	Description	Example

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

10.1.6.8.2 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", } } }
```

10.1.6.9 Verification

See for more information on the source of these fields.

10.1.6.9.1 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" } } }
```

10.1.6.9.2 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 } } }
```

10.1.6.9.3 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" } } }
```

10.1.6.9.4 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 } } }
```

10.1.6.10 Cookies

10.1.6.10.1 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" } } ] }
```

10.1.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

10.1.6.11 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.

10.1.6.11.1 Documentation request

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

10.1.6.11.2 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" ] }
```

10.1.6.12 i18 internationalization

 # Retrieves i18next json based on requested language

10.1.6.12.1 i18 request

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

11 Backticks

11.1 Backticks MD file

this is single line of backticks

Tis is text which contains text under backticks

this is backticks this is regular text

several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks

11.1.1 *Backtickts in heading*

test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test

11.2 End of file

test

12 Backticks

12.1 Backticks MD file

this is single line of backticks

Tis is text which contains text under backticks

this is backticks this is regular text

several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks several text lines under single backticks

12.1.1 *Backtickts in heading*

test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test test

12.2 End of file

test

13 Backticks

13.1 Backticks MD file

this is single line of backticks

Tis is text which contains text under backticks

this is backticks this is regular text

several text lines under single backticks several text lines under single backticks several text lines under single backticks

14 Just checking tables - heading 1

14.1 Tables are under heading 2

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

Table

Option	Description
data	path to data files to supply the data that will be passed into templates.
engine	engine to be used for processing templates. Handlebars is the default.
ext	extension to be used for dest files.

Table

15 test2

15.1 Markdown2-2markdown2-2markdown2-2markdown2-2markdown2-2markdown2-2markdown2-2markdown2-2ma

15.1.1 Formatted Text Examples

Below is an illustration of the Format toolbar used to format text:

Here are some examples of the effects different options have on text:

15.1.1.1 Font buttons:

As well as the standard Font drop down box showing a list of all the fonts on your system and the **Point** size box that allows you to set the font size, two shortcut buttons have been added that allow you to increase the selected text or group of text objects at the press of a button:

Will increase the font size by 2 points every time you press the button.

Will decrease the font size by 2 points every time you press the button.

15.1.1.2 Text Format:

The standard text format options are also available for text i.e. bold, italic, underline and outline:

Text

Text

Text

Text

> ***Note: You can combine any of these formats together, as in the last example above which is a combination of the bold, underline and outline formats.***

15.1.1.3 Text Justification:

Text can be aligned to the
    1. *left,*
    2. *right,*
    3. *centre*
or one of the three full justifications: Full Justification, Density Justification and Aspect Justification.

Note: The Full justification spreads the words across the page but does not change the size of them. The Density Justification also spreads the words and does not change their size but puts spaces between the characters in a word. The Aspect Justification spreads the words but stretches the words to fill the space as well.

The Density and Aspect methods can provide some visually interesting effects.

Go back to the top

15.1.2 Useful markdown links:

Link 1
    1. *This is link 1*
Link 2
    1. *This is link 2*
Link 3
    1. *This is link 3*
        # *Link 4*
            # *This is link 4 ordered under link 3*

15.1.2.1 I just have to store some code here, if you don't mind

15.1.2.1.1 I'll put it here

LoGiC *iS* **rEaL** *!!!*

```
if(a == 0){  a = 0; }
```

15.1.2.1.2 And also here

It's getting better

```
if (false == true) {  // log some message }
```

15.1.2.1.3 You know the process...

I'll just assert and finish this...

```
public void justAssert(){  boolean checkTrue = true;  Assert.assertTrue(checkTrue); }  //let me return true he he  public boolean trueStatement()
{  return true; }
```

15.2 Markdown2-2markdown2-2markdown2-2markdown2-2markdown2-2markdown2-2markdown2-2markdown2-2ma

15.2.1 Formatted Text Examples

Below is an illustration of the Format toolbar used to format text:

Here are some examples of the effects different options have on text:

15.2.1.1 Font buttons:

As well as the standard Font drop down box showing a list of all the fonts on your system and the **Point** size box that allows you to set the font size, two shortcut buttons have been added that allow you to increase the selected text or group of text objects at the press of a button:

Will increase the font size by 2 points every time you press the button.

Will decrease the font size by 2 points every time you press the button.

15.2.1.2 Text Format:

The standard text format options are also available for text i.e. bold, italic, underline and outline:

> ***Note: You can combine any of these formats together, as in the last example above which is a combination of the bold, underline and outline formats.***

15.2.1.3 Text Justification:

- *Text can be aligned to the*
  1. *left,*
  2. *right,*
  3. *centre*
- *or one of the three full justifications: Full Justification, Density Justification and Aspect Justification.*

Note: The Full justification spreads the words across the page but does not change the size of them. The Density Justification also spreads the words and does not change their size but puts spaces between the characters in a word. The Aspect Justification spreads the words but stretches the words to fill the space as well.

The Density and Aspect methods can provide some visually interesting effects.

Go back to the top

15.2.2 Useful markdown links:

- *Link 1*
  1. *This is link 1*

Link 2

1. *This is link 2*

Link 3

1. *This is link 3*

Link 4

This is link 4 ordered under link 3

15.2.2.1 I just have to store some code here, if you don't mind

15.2.2.1.1 I'll put it here

LoGiC *iS* **rEaL** *!!!*

```
if(a == 0){  a = 0; }
```

15.2.2.1.2 And also here

It's getting better

```
if (false == true) {  // log some message }
```

15.2.2.1.3 You know the process...

I'll just assert and finish this...

```
public void justAssert(){  boolean checkTrue = true;  Assert.assertTrue(checkTrue); }  //let me return true he he  public boolean trueStatement()
{  return true; }
```

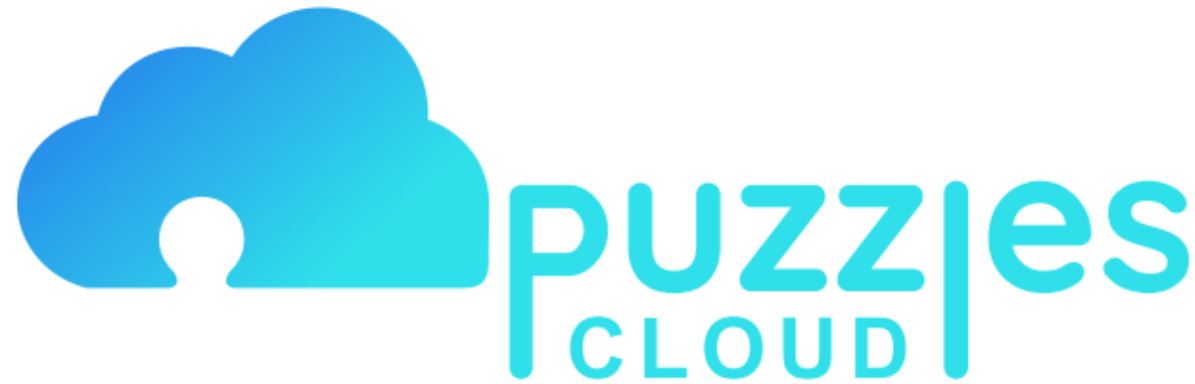


Figure 1: Last Page