



We enable docx-as-code

Advanced Template version: 1.0

[[doc-title]]

[[doc-subtitle]]

[[doc-type]]

[[doc-company]]

[[doc-company-url]]

Document Number: [[doc-id]]

Version: [[doc-version]]

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



We enable docx-as-code

Document History

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

Document Revision

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

Intellectual Property Rights

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



Table of Contents

1 Introduction.....	25
1.1 References.....	25
2 Big.....	26
2.1 Edge Service.....	26
2.1.1 TODO.....	26
2.1.1.1 TODO version 1.0.....	26
2.1.1.2 TODO version 1.0-next.....	26
2.1.2 Test environment and credentials.....	26
2.1.3 Edge Service Registry.....	26
2.1.4 Edge Service Relay.....	26
2.1.5 Edge Service Client.....	27
2.1.6 ReST API.....	27
2.1.6.1 Login.....	27
2.1.6.1.1 Login request.....	27
2.1.6.1.2 Login response.....	27
2.1.6.1.3 Cookies.....	27
2.1.6.2 Logout.....	27
2.1.6.2.1 Logout request.....	27
2.1.6.2.2 Logout response.....	28
2.1.6.3 Voucher list.....	28
2.1.6.3.1 Voucher list request.....	28
2.1.6.3.2 Voucher list response.....	28
2.1.6.4 Voucher detail.....	28
2.1.6.4.1 Voucher detail request.....	29
2.1.6.4.2 Voucher detail response.....	29
2.1.6.5 Voucher order.....	29
2.1.6.5.1 Voucher order request.....	29
2.1.6.5.2 Voucher order response.....	30
2.1.6.6 Voucher order detail.....	30
2.1.6.6.1 Voucher order detail request.....	31
2.1.6.6.2 Voucher order detail response.....	31
2.1.6.7 Provider Text.....	31
2.1.6.7.1 Provider Text request.....	31
2.1.6.7.2 Provider Text response.....	31
2.1.6.8 Content.....	31
2.1.6.8.1 Content request.....	31
2.1.6.8.2 Content response.....	32
2.1.6.9 Verification.....	32
2.1.6.9.1 Verification request by phone number.....	32
2.1.6.9.2 Verification by phone number response.....	32
2.1.6.9.3 Verification request by sms.....	32
2.1.6.9.4 Verification by sms number response.....	32
2.1.6.10 Cookies.....	32
2.1.6.10.1 Cookies request.....	32
2.1.6.10.2 Cookies response.....	32
2.1.6.11 Documentation.....	32
2.1.6.11.1 Documentation request.....	33
2.1.6.11.2 Documentation response.....	33
2.1.6.12 i18n internationalization.....	33
2.1.6.12.1 i18n request.....	33



2.2 Edge Service.....	33
2.2.1 TODO.....	33
2.2.1.1 TODO version 1.0.....	33
2.2.1.2 TODO version 1.0-next.....	33
2.2.2 Test environment and credentials.....	33
2.2.3 Edge Service Registry.....	33
2.2.4 Edge Service Relay.....	34
2.2.5 Edge Service Client.....	34
2.2.6 ReST API.....	34
2.2.6.1 Login.....	34
2.2.6.1.1 Login request.....	34
2.2.6.1.2 Login response.....	35
2.2.6.1.3 Cookies.....	35
2.2.6.2 Logout.....	35
2.2.6.2.1 Logout request.....	35
2.2.6.2.2 Logout response.....	35
2.2.6.3 Voucher list.....	35
2.2.6.3.1 Voucher list request.....	36
2.2.6.3.2 Voucher list response.....	36
2.2.6.4 Voucher detail.....	36
2.2.6.4.1 Voucher detail request.....	36
2.2.6.4.2 Voucher detail response.....	36
2.2.6.5 Voucher order.....	36
2.2.6.5.1 Voucher order request.....	36
2.2.6.5.2 Voucher order response.....	38
2.2.6.6 Voucher order detail.....	38
2.2.6.6.1 Voucher order detail request.....	38
2.2.6.6.2 Voucher order detail response.....	38
2.2.6.7 Provider Text.....	38
2.2.6.7.1 Provider Text request.....	38
2.2.6.7.2 Provider Text response.....	39
2.2.6.8 Content.....	39
2.2.6.8.1 Content request.....	39
2.2.6.8.2 Content response.....	39
2.2.6.9 Verification.....	39
2.2.6.9.1 Verification request by phone number.....	40
2.2.6.9.2 Verification by phone number response.....	40
2.2.6.9.3 Verification request by sms.....	40
2.2.6.9.4 Verification by sms number response.....	40
2.2.6.10 Cookies.....	40
2.2.6.10.1 Cookies request.....	40
2.2.6.10.2 Cookies response.....	40
2.2.6.11 Documentation.....	40
2.2.6.11.1 Documentation request.....	40
2.2.6.11.2 Documentation response.....	40
2.2.6.12 i18 internationalization.....	41
2.2.6.12.1 i18 request.....	41
2.3 Edge Service.....	41
2.3.1 TODO.....	41
2.3.1.1 TODO version 1.0.....	41
2.3.1.2 TODO version 1.0-next.....	41
2.3.2 Test environment and credentials.....	41
2.3.3 Edge Service Registry.....	41
2.3.4 Edge Service Relay.....	42
2.3.5 Edge Service Client.....	42



2.3.6 ReST API.....	42
2.3.6.1 Login.....	42
2.3.6.1.1 Login request.....	42
2.3.6.1.2 Login response.....	43
2.3.6.1.3 Cookies.....	43
2.3.6.2 Logout.....	43
2.3.6.2.1 Logout request.....	43
2.3.6.2.2 Logout response.....	43
2.3.6.3 Voucher list.....	43
2.3.6.3.1 Voucher list request.....	44
2.3.6.3.2 Voucher list response.....	44
2.3.6.4 Voucher detail.....	44
2.3.6.4.1 Voucher detail request.....	44
2.3.6.4.2 Voucher detail response.....	44
2.3.6.5 Voucher order.....	44
2.3.6.5.1 Voucher order request.....	44
2.3.6.5.2 Voucher order response.....	46
2.3.6.6 Voucher order detail.....	46
2.3.6.6.1 Voucher order detail request.....	46
2.3.6.6.2 Voucher order detail response.....	46
2.3.6.7 Provider Text.....	46
2.3.6.7.1 Provider Text request.....	46
2.3.6.7.2 Provider Text response.....	47
2.3.6.8 Content.....	47
2.3.6.8.1 Content request.....	47
2.3.6.8.2 Content response.....	47
2.3.6.9 Verification.....	47
2.3.6.9.1 Verification request by phone number.....	48
2.3.6.9.2 Verification by phone number response.....	48
2.3.6.9.3 Verification request by sms.....	48
2.3.6.9.4 Verification by sms number response.....	48
2.3.6.10 Cookies.....	48
2.3.6.10.1 Cookies request.....	48
2.3.6.10.2 Cookies response.....	48
2.3.6.11 Documentation.....	48
2.3.6.11.1 Documentation request.....	48
2.3.6.11.2 Documentation response.....	48
2.3.6.12 i18n internationalization.....	49
2.3.6.12.1 i18n request.....	49
2.4 Edge Service.....	49
2.4.1 TODO.....	49
2.4.1.1 TODO version 1.0.....	49
2.4.1.2 TODO version 1.0-next.....	49
2.4.2 Test environment and credentials.....	49
2.4.3 Edge Service Registry.....	49
2.4.4 Edge Service Relay.....	50
2.4.5 Edge Service Client.....	50
2.4.6 ReST API.....	50
2.4.6.1 Login.....	50
2.4.6.1.1 Login request.....	50
2.4.6.1.2 Login response.....	51
2.4.6.1.3 Cookies.....	51
2.4.6.2 Logout.....	51
2.4.6.2.1 Logout request.....	51
2.4.6.2.2 Logout response.....	51



2.4.6.3 Voucher list.....	51
2.4.6.3.1 Voucher list request.....	52
2.4.6.3.2 Voucher list response.....	52
2.4.6.4 Voucher detail.....	52
2.4.6.4.1 Voucher detail request.....	52
2.4.6.4.2 Voucher detail response.....	52
2.4.6.5 Voucher order.....	52
2.4.6.5.1 Voucher order request.....	52
2.4.6.5.2 Voucher order response.....	54
2.4.6.6 Voucher order detail.....	54
2.4.6.6.1 Voucher order detail request.....	54
2.4.6.6.2 Voucher order detail response.....	54
2.4.6.7 Provider Text.....	54
2.4.6.7.1 Provider Text request.....	54
2.4.6.7.2 Provider Text response.....	55
2.4.6.8 Content.....	55
2.4.6.8.1 Content request.....	55
2.4.6.8.2 Content response.....	55
2.4.6.9 Verification.....	55
2.4.6.9.1 Verification request by phone number.....	56
2.4.6.9.2 Verification by phone number response.....	56
2.4.6.9.3 Verification request by sms.....	56
2.4.6.9.4 Verification by sms number response.....	56
2.4.6.10 Cookies.....	56
2.4.6.10.1 Cookies request.....	56
2.4.6.10.2 Cookies response.....	56
2.4.6.11 Documentation.....	56
2.4.6.11.1 Documentation request.....	56
2.4.6.11.2 Documentation response.....	56
2.4.6.12 i18n internationalization.....	57
2.4.6.12.1 i18n request.....	57
2.5 Edge Service.....	57
2.5.1 TODO.....	57
2.5.1.1 TODO version 1.0.....	57
2.5.1.2 TODO version 1.0-next.....	57
2.5.2 Test environment and credentials.....	57
2.5.3 Edge Service Registry.....	57
2.5.4 Edge Service Relay.....	58
2.5.5 Edge Service Client.....	58
2.5.6 ReST API.....	58
2.5.6.1 Login.....	58
2.5.6.1.1 Login request.....	58
2.5.6.1.2 Login response.....	59
2.5.6.1.3 Cookies.....	59
2.5.6.2 Logout.....	59
2.5.6.2.1 Logout request.....	59
2.5.6.2.2 Logout response.....	59
2.5.6.3 Voucher list.....	59
2.5.6.3.1 Voucher list request.....	60
2.5.6.3.2 Voucher list response.....	60
2.5.6.4 Voucher detail.....	60
2.5.6.4.1 Voucher detail request.....	60
2.5.6.4.2 Voucher detail response.....	60
2.5.6.5 Voucher order.....	60
2.5.6.5.1 Voucher order request.....	60



2.5.6.5.2 Voucher order response.....	62
2.5.6.6 Voucher order detail.....	62
2.5.6.6.1 Voucher order detail request.....	62
2.5.6.6.2 Voucher order detail response.....	62
2.5.6.7 Provider Text.....	62
2.5.6.7.1 Provider Text request.....	62
2.5.6.7.2 Provider Text response.....	63
2.5.6.8 Content.....	63
2.5.6.8.1 Content request.....	63
2.5.6.8.2 Content response.....	63
2.5.6.9 Verification.....	63
2.5.6.9.1 Verification request by phone number.....	64
2.5.6.9.2 Verification by phone number response.....	64
2.5.6.9.3 Verification request by sms.....	64
2.5.6.9.4 Verification by sms number response.....	64
2.5.6.10 Cookies.....	64
2.5.6.10.1 Cookies request.....	64
2.5.6.10.2 Cookies response.....	64
2.5.6.11 Documentation.....	64
2.5.6.11.1 Documentation request.....	64
2.5.6.11.2 Documentation response.....	64
2.5.6.12 i18n internationalization.....	65
2.5.6.12.1 i18n request.....	65
3 Big.....	66
3.1 Edge Service.....	66
3.1.1 TODO.....	66
3.1.1.1 TODO version 1.0.....	66
3.1.1.2 TODO version 1.0-next.....	66
3.1.2 Test environment and credentials.....	66
3.1.3 Edge Service Registry.....	66
3.1.4 Edge Service Relay.....	66
3.1.5 Edge Service Client.....	67
3.1.6 ReST API.....	67
3.1.6.1 Login.....	67
3.1.6.1.1 Login request.....	67
3.1.6.1.2 Login response.....	67
3.1.6.1.3 Cookies.....	67
3.1.6.2 Logout.....	67
3.1.6.2.1 Logout request.....	67
3.1.6.2.2 Logout response.....	68
3.1.6.3 Voucher list.....	68
3.1.6.3.1 Voucher list request.....	68
3.1.6.3.2 Voucher list response.....	68
3.1.6.4 Voucher detail.....	68
3.1.6.4.1 Voucher detail request.....	69
3.1.6.4.2 Voucher detail response.....	69
3.1.6.5 Voucher order.....	69
3.1.6.5.1 Voucher order request.....	69
3.1.6.5.2 Voucher order response.....	70
3.1.6.6 Voucher order detail.....	70
3.1.6.6.1 Voucher order detail request.....	71
3.1.6.6.2 Voucher order detail response.....	71
3.1.6.7 Provider Text.....	71
3.1.6.7.1 Provider Text request.....	71



3.1.6.7.2 Provider Text response.....	71
3.1.6.8 Content.....	71
3.1.6.8.1 Content request.....	71
3.1.6.8.2 Content response.....	72
3.1.6.9 Verification.....	72
3.1.6.9.1 Verification request by phone number.....	72
3.1.6.9.2 Verification by phone number response.....	72
3.1.6.9.3 Verification request by sms.....	72
3.1.6.9.4 Verification by sms number response.....	72
3.1.6.10 Cookies.....	72
3.1.6.10.1 Cookies request.....	72
3.1.6.10.2 Cookies response.....	72
3.1.6.11 Documentation.....	72
3.1.6.11.1 Documentation request.....	73
3.1.6.11.2 Documentation response.....	73
3.1.6.12 i18n internationalization.....	73
3.1.6.12.1 i18n request.....	73
3.2 Edge Service.....	73
3.2.1 TODO.....	73
3.2.1.1 TODO version 1.0.....	73
3.2.1.2 TODO version 1.0-next.....	73
3.2.2 Test environment and credentials.....	73
3.2.3 Edge Service Registry.....	73
3.2.4 Edge Service Relay.....	74
3.2.5 Edge Service Client.....	74
3.2.6 ReST API.....	74
3.2.6.1 Login.....	74
3.2.6.1.1 Login request.....	74
3.2.6.1.2 Login response.....	75
3.2.6.1.3 Cookies.....	75
3.2.6.2 Logout.....	75
3.2.6.2.1 Logout request.....	75
3.2.6.2.2 Logout response.....	75
3.2.6.3 Voucher list.....	75
3.2.6.3.1 Voucher list request.....	76
3.2.6.3.2 Voucher list response.....	76
3.2.6.4 Voucher detail.....	76
3.2.6.4.1 Voucher detail request.....	76
3.2.6.4.2 Voucher detail response.....	76
3.2.6.5 Voucher order.....	76
3.2.6.5.1 Voucher order request.....	76
3.2.6.5.2 Voucher order response.....	78
3.2.6.6 Voucher order detail.....	78
3.2.6.6.1 Voucher order detail request.....	78
3.2.6.6.2 Voucher order detail response.....	78
3.2.6.7 Provider Text.....	78
3.2.6.7.1 Provider Text request.....	78
3.2.6.7.2 Provider Text response.....	79
3.2.6.8 Content.....	79
3.2.6.8.1 Content request.....	79
3.2.6.8.2 Content response.....	79
3.2.6.9 Verification.....	79
3.2.6.9.1 Verification request by phone number.....	80
3.2.6.9.2 Verification by phone number response.....	80
3.2.6.9.3 Verification request by sms.....	80



3.2.6.9.4 Verification by sms number response.....	80
3.2.6.10 Cookies.....	80
3.2.6.10.1 Cookies request.....	80
3.2.6.10.2 Cookies response.....	80
3.2.6.11 Documentation.....	80
3.2.6.11.1 Documentation request.....	80
3.2.6.11.2 Documentation response.....	80
3.2.6.12 i18 internationalization.....	81
3.2.6.12.1 i18 request.....	81
3.3 Edge Service.....	81
3.3.1 TODO.....	81
3.3.1.1 TODO version 1.0.....	81
3.3.1.2 TODO version 1.0-next.....	81
3.3.2 Test environment and credentials.....	81
3.3.3 Edge Service Registry.....	81
3.3.4 Edge Service Relay.....	82
3.3.5 Edge Service Client.....	82
3.3.6 ReST API.....	82
3.3.6.1 Login.....	82
3.3.6.1.1 Login request.....	82
3.3.6.1.2 Login response.....	83
3.3.6.1.3 Cookies.....	83
3.3.6.2 Logout.....	83
3.3.6.2.1 Logout request.....	83
3.3.6.2.2 Logout response.....	83
3.3.6.3 Voucher list.....	83
3.3.6.3.1 Voucher list request.....	84
3.3.6.3.2 Voucher list response.....	84
3.3.6.4 Voucher detail.....	84
3.3.6.4.1 Voucher detail request.....	84
3.3.6.4.2 Voucher detail response.....	84
3.3.6.5 Voucher order.....	84
3.3.6.5.1 Voucher order request.....	84
3.3.6.5.2 Voucher order response.....	86
3.3.6.6 Voucher order detail.....	86
3.3.6.6.1 Voucher order detail request.....	86
3.3.6.6.2 Voucher order detail response.....	86
3.3.6.7 Provider Text.....	86
3.3.6.7.1 Provider Text request.....	86
3.3.6.7.2 Provider Text response.....	87
3.3.6.8 Content.....	87
3.3.6.8.1 Content request.....	87
3.3.6.8.2 Content response.....	87
3.3.6.9 Verification.....	87
3.3.6.9.1 Verification request by phone number.....	88
3.3.6.9.2 Verification by phone number response.....	88
3.3.6.9.3 Verification request by sms.....	88
3.3.6.9.4 Verification by sms number response.....	88
3.3.6.10 Cookies.....	88
3.3.6.10.1 Cookies request.....	88
3.3.6.10.2 Cookies response.....	88
3.3.6.11 Documentation.....	88
3.3.6.11.1 Documentation request.....	88
3.3.6.11.2 Documentation response.....	88
3.3.6.12 i18 internationalization.....	89



3.3.6.12.1 i18 request.....	89
3.4 Edge Service.....	89
3.4.1 TODO.....	89
3.4.1.1 TODO version 1.0.....	89
3.4.1.2 TODO version 1.0-next.....	89
3.4.2 Test environment and credentials.....	89
3.4.3 Edge Service Registry.....	89
3.4.4 Edge Service Relay.....	90
3.4.5 Edge Service Client.....	90
3.4.6 ReST API.....	90
3.4.6.1 Login.....	90
3.4.6.1.1 Login request.....	90
3.4.6.1.2 Login response.....	91
3.4.6.1.3 Cookies.....	91
3.4.6.2 Logout.....	91
3.4.6.2.1 Logout request.....	91
3.4.6.2.2 Logout response.....	91
3.4.6.3 Voucher list.....	91
3.4.6.3.1 Voucher list request.....	92
3.4.6.3.2 Voucher list response.....	92
3.4.6.4 Voucher detail.....	92
3.4.6.4.1 Voucher detail request.....	92
3.4.6.4.2 Voucher detail response.....	92
3.4.6.5 Voucher order.....	92
3.4.6.5.1 Voucher order request.....	92
3.4.6.5.2 Voucher order response.....	94
3.4.6.6 Voucher order detail.....	94
3.4.6.6.1 Voucher order detail request.....	94
3.4.6.6.2 Voucher order detail response.....	94
3.4.6.7 Provider Text.....	94
3.4.6.7.1 Provider Text request.....	94
3.4.6.7.2 Provider Text response.....	95
3.4.6.8 Content.....	95
3.4.6.8.1 Content request.....	95
3.4.6.8.2 Content response.....	95
3.4.6.9 Verification.....	95
3.4.6.9.1 Verification request by phone number.....	96
3.4.6.9.2 Verification by phone number response.....	96
3.4.6.9.3 Verification request by sms.....	96
3.4.6.9.4 Verification by sms number response.....	96
3.4.6.10 Cookies.....	96
3.4.6.10.1 Cookies request.....	96
3.4.6.10.2 Cookies response.....	96
3.4.6.11 Documentation.....	96
3.4.6.11.1 Documentation request.....	96
3.4.6.11.2 Documentation response.....	96
3.4.6.12 i18 internationalization.....	97
3.4.6.12.1 i18 request.....	97
3.5 Edge Service.....	97
3.5.1 TODO.....	97
3.5.1.1 TODO version 1.0.....	97
3.5.1.2 TODO version 1.0-next.....	97
3.5.2 Test environment and credentials.....	97
3.5.3 Edge Service Registry.....	97



3.5.4 Edge Service Relay.....	98
3.5.5 Edge Service Client.....	98
3.5.6 ReST API.....	98
3.5.6.1 Login.....	98
3.5.6.1.1 Login request.....	98
3.5.6.1.2 Login response.....	99
3.5.6.1.3 Cookies.....	99
3.5.6.2 Logout.....	99
3.5.6.2.1 Logout request.....	99
3.5.6.2.2 Logout response.....	99
3.5.6.3 Voucher list.....	99
3.5.6.3.1 Voucher list request.....	100
3.5.6.3.2 Voucher list response.....	100
3.5.6.4 Voucher detail.....	100
3.5.6.4.1 Voucher detail request.....	100
3.5.6.4.2 Voucher detail response.....	100
3.5.6.5 Voucher order.....	100
3.5.6.5.1 Voucher order request.....	100
3.5.6.5.2 Voucher order response.....	102
3.5.6.6 Voucher order detail.....	102
3.5.6.6.1 Voucher order detail request.....	102
3.5.6.6.2 Voucher order detail response.....	102
3.5.6.7 Provider Text.....	102
3.5.6.7.1 Provider Text request.....	102
3.5.6.7.2 Provider Text response.....	103
3.5.6.8 Content.....	103
3.5.6.8.1 Content request.....	103
3.5.6.8.2 Content response.....	103
3.5.6.9 Verification.....	103
3.5.6.9.1 Verification request by phone number.....	104
3.5.6.9.2 Verification by phone number response.....	104
3.5.6.9.3 Verification request by sms.....	104
3.5.6.9.4 Verification by sms number response.....	104
3.5.6.10 Cookies.....	104
3.5.6.10.1 Cookies request.....	104
3.5.6.10.2 Cookies response.....	104
3.5.6.11 Documentation.....	104
3.5.6.11.1 Documentation request.....	104
3.5.6.11.2 Documentation response.....	104
3.5.6.12 i18n internationalization.....	105
3.5.6.12.1 i18n request.....	105
4 Big.....	106
4.1 Edge Service.....	106
4.1.1 TODO.....	106
4.1.1.1 TODO version 1.0.....	106
4.1.1.2 TODO version 1.0-next.....	106
4.1.2 Test environment and credentials.....	106
4.1.3 Edge Service Registry.....	106
4.1.4 Edge Service Relay.....	106
4.1.5 Edge Service Client.....	107
4.1.6 ReST API.....	107
4.1.6.1 Login.....	107
4.1.6.1.1 Login request.....	107
4.1.6.1.2 Login response.....	107



4.1.6.1.3 Cookies.....	107
4.1.6.2 Logout.....	107
4.1.6.2.1 Logout request.....	107
4.1.6.2.2 Logout response.....	108
4.1.6.3 Voucher list.....	108
4.1.6.3.1 Voucher list request.....	108
4.1.6.3.2 Voucher list response.....	108
4.1.6.4 Voucher detail.....	108
4.1.6.4.1 Voucher detail request.....	109
4.1.6.4.2 Voucher detail response.....	109
4.1.6.5 Voucher order.....	109
4.1.6.5.1 Voucher order request.....	109
4.1.6.5.2 Voucher order response.....	110
4.1.6.6 Voucher order detail.....	110
4.1.6.6.1 Voucher order detail request.....	111
4.1.6.6.2 Voucher order detail response.....	111
4.1.6.7 Provider Text.....	111
4.1.6.7.1 Provider Text request.....	111
4.1.6.7.2 Provider Text response.....	111
4.1.6.8 Content.....	111
4.1.6.8.1 Content request.....	111
4.1.6.8.2 Content response.....	112
4.1.6.9 Verification.....	112
4.1.6.9.1 Verification request by phone number.....	112
4.1.6.9.2 Verification by phone number response.....	112
4.1.6.9.3 Verification request by sms.....	112
4.1.6.9.4 Verification by sms number response.....	112
4.1.6.10 Cookies.....	112
4.1.6.10.1 Cookies request.....	112
4.1.6.10.2 Cookies response.....	112
4.1.6.11 Documentation.....	112
4.1.6.11.1 Documentation request.....	113
4.1.6.11.2 Documentation response.....	113
4.1.6.12 i18n internationalization.....	113
4.1.6.12.1 i18n request.....	113
4.2 Edge Service.....	113
4.2.1 TODO.....	113
4.2.1.1 TODO version 1.0.....	113
4.2.1.2 TODO version 1.0-next.....	113
4.2.2 Test environment and credentials.....	113
4.2.3 Edge Service Registry.....	113
4.2.4 Edge Service Relay.....	114
4.2.5 Edge Service Client.....	114
4.2.6 ReST API.....	114
4.2.6.1 Login.....	114
4.2.6.1.1 Login request.....	114
4.2.6.1.2 Login response.....	115
4.2.6.1.3 Cookies.....	115
4.2.6.2 Logout.....	115
4.2.6.2.1 Logout request.....	115
4.2.6.2.2 Logout response.....	115
4.2.6.3 Voucher list.....	115
4.2.6.3.1 Voucher list request.....	116
4.2.6.3.2 Voucher list response.....	116
4.2.6.4 Voucher detail.....	116



4.2.6.4.1 Voucher detail request.....	116
4.2.6.4.2 Voucher detail response.....	116
4.2.6.5 Voucher order.....	116
4.2.6.5.1 Voucher order request.....	116
4.2.6.5.2 Voucher order response.....	118
4.2.6.6 Voucher order detail.....	118
4.2.6.6.1 Voucher order detail request.....	118
4.2.6.6.2 Voucher order detail response.....	118
4.2.6.7 Provider Text.....	118
4.2.6.7.1 Provider Text request.....	118
4.2.6.7.2 Provider Text response.....	119
4.2.6.8 Content.....	119
4.2.6.8.1 Content request.....	119
4.2.6.8.2 Content response.....	119
4.2.6.9 Verification.....	119
4.2.6.9.1 Verification request by phone number.....	120
4.2.6.9.2 Verification by phone number response.....	120
4.2.6.9.3 Verification request by sms.....	120
4.2.6.9.4 Verification by sms number response.....	120
4.2.6.10 Cookies.....	120
4.2.6.10.1 Cookies request.....	120
4.2.6.10.2 Cookies response.....	120
4.2.6.11 Documentation.....	120
4.2.6.11.1 Documentation request.....	120
4.2.6.11.2 Documentation response.....	120
4.2.6.12 i18n internationalization.....	121
4.2.6.12.1 i18n request.....	121
4.3 Edge Service.....	121
4.3.1 TODO.....	121
4.3.1.1 TODO version 1.0.....	121
4.3.1.2 TODO version 1.0-next.....	121
4.3.2 Test environment and credentials.....	121
4.3.3 Edge Service Registry.....	121
4.3.4 Edge Service Relay.....	122
4.3.5 Edge Service Client.....	122
4.3.6 ReST API.....	122
4.3.6.1 Login.....	122
4.3.6.1.1 Login request.....	122
4.3.6.1.2 Login response.....	123
4.3.6.1.3 Cookies.....	123
4.3.6.2 Logout.....	123
4.3.6.2.1 Logout request.....	123
4.3.6.2.2 Logout response.....	123
4.3.6.3 Voucher list.....	123
4.3.6.3.1 Voucher list request.....	124
4.3.6.3.2 Voucher list response.....	124
4.3.6.4 Voucher detail.....	124
4.3.6.4.1 Voucher detail request.....	124
4.3.6.4.2 Voucher detail response.....	124
4.3.6.5 Voucher order.....	124
4.3.6.5.1 Voucher order request.....	124
4.3.6.5.2 Voucher order response.....	126
4.3.6.6 Voucher order detail.....	126
4.3.6.6.1 Voucher order detail request.....	126
4.3.6.6.2 Voucher order detail response.....	126



4.3.6.7 Provider Text.....	126
4.3.6.7.1 Provider Text request.....	126
4.3.6.7.2 Provider Text response.....	127
4.3.6.8 Content.....	127
4.3.6.8.1 Content request.....	127
4.3.6.8.2 Content response.....	127
4.3.6.9 Verification.....	127
4.3.6.9.1 Verification request by phone number.....	128
4.3.6.9.2 Verification by phone number response.....	128
4.3.6.9.3 Verification request by sms.....	128
4.3.6.9.4 Verification by sms number response.....	128
4.3.6.10 Cookies.....	128
4.3.6.10.1 Cookies request.....	128
4.3.6.10.2 Cookies response.....	128
4.3.6.11 Documentation.....	128
4.3.6.11.1 Documentation request.....	128
4.3.6.11.2 Documentation response.....	128
4.3.6.12 i18n internationalization.....	129
4.3.6.12.1 i18n request.....	129
4.4 Edge Service.....	129
4.4.1 TODO.....	129
4.4.1.1 TODO version 1.0.....	129
4.4.1.2 TODO version 1.0-next.....	129
4.4.2 Test environment and credentials.....	129
4.4.3 Edge Service Registry.....	129
4.4.4 Edge Service Relay.....	130
4.4.5 Edge Service Client.....	130
4.4.6 ReST API.....	130
4.4.6.1 Login.....	130
4.4.6.1.1 Login request.....	130
4.4.6.1.2 Login response.....	131
4.4.6.1.3 Cookies.....	131
4.4.6.2 Logout.....	131
4.4.6.2.1 Logout request.....	131
4.4.6.2.2 Logout response.....	131
4.4.6.3 Voucher list.....	131
4.4.6.3.1 Voucher list request.....	132
4.4.6.3.2 Voucher list response.....	132
4.4.6.4 Voucher detail.....	132
4.4.6.4.1 Voucher detail request.....	132
4.4.6.4.2 Voucher detail response.....	132
4.4.6.5 Voucher order.....	132
4.4.6.5.1 Voucher order request.....	132
4.4.6.5.2 Voucher order response.....	134
4.4.6.6 Voucher order detail.....	134
4.4.6.6.1 Voucher order detail request.....	134
4.4.6.6.2 Voucher order detail response.....	134
4.4.6.7 Provider Text.....	134
4.4.6.7.1 Provider Text request.....	134
4.4.6.7.2 Provider Text response.....	135
4.4.6.8 Content.....	135
4.4.6.8.1 Content request.....	135
4.4.6.8.2 Content response.....	135
4.4.6.9 Verification.....	135
4.4.6.9.1 Verification request by phone number.....	136



4.4.6.9.2 Verification by phone number response.....	136
4.4.6.9.3 Verification request by sms.....	136
4.4.6.9.4 Verification by sms number response.....	136
4.4.6.10 Cookies.....	136
4.4.6.10.1 Cookies request.....	136
4.4.6.10.2 Cookies response.....	136
4.4.6.11 Documentation.....	136
4.4.6.11.1 Documentation request.....	136
4.4.6.11.2 Documentation response.....	136
4.4.6.12 i18n internationalization.....	137
4.4.6.12.1 i18n request.....	137
4.5 Edge Service.....	137
4.5.1 TODO.....	137
4.5.1.1 TODO version 1.0.....	137
4.5.1.2 TODO version 1.0-next.....	137
4.5.2 Test environment and credentials.....	137
4.5.3 Edge Service Registry.....	137
4.5.4 Edge Service Relay.....	138
4.5.5 Edge Service Client.....	138
4.5.6 ReST API.....	138
4.5.6.1 Login.....	138
4.5.6.1.1 Login request.....	138
4.5.6.1.2 Login response.....	139
4.5.6.1.3 Cookies.....	139
4.5.6.2 Logout.....	139
4.5.6.2.1 Logout request.....	139
4.5.6.2.2 Logout response.....	139
4.5.6.3 Voucher list.....	139
4.5.6.3.1 Voucher list request.....	140
4.5.6.3.2 Voucher list response.....	140
4.5.6.4 Voucher detail.....	140
4.5.6.4.1 Voucher detail request.....	140
4.5.6.4.2 Voucher detail response.....	140
4.5.6.5 Voucher order.....	140
4.5.6.5.1 Voucher order request.....	140
4.5.6.5.2 Voucher order response.....	142
4.5.6.6 Voucher order detail.....	142
4.5.6.6.1 Voucher order detail request.....	142
4.5.6.6.2 Voucher order detail response.....	142
4.5.6.7 Provider Text.....	142
4.5.6.7.1 Provider Text request.....	142
4.5.6.7.2 Provider Text response.....	143
4.5.6.8 Content.....	143
4.5.6.8.1 Content request.....	143
4.5.6.8.2 Content response.....	143
4.5.6.9 Verification.....	143
4.5.6.9.1 Verification request by phone number.....	144
4.5.6.9.2 Verification by phone number response.....	144
4.5.6.9.3 Verification request by sms.....	144
4.5.6.9.4 Verification by sms number response.....	144
4.5.6.10 Cookies.....	144
4.5.6.10.1 Cookies request.....	144
4.5.6.10.2 Cookies response.....	144
4.5.6.11 Documentation.....	144
4.5.6.11.1 Documentation request.....	144



4.5.6.11.2 Documentation response.....	144
4.5.6.12 i18 internationalization.....	145
4.5.6.12.1 i18 request.....	145
5 Big.....	146
5.1 Edge Service.....	146
5.1.1 TODO.....	146
5.1.1.1 TODO version 1.0.....	146
5.1.1.2 TODO version 1.0-next.....	146
5.1.2 Test environment and credentials.....	146
5.1.3 Edge Service Registry.....	146
5.1.4 Edge Service Relay.....	146
5.1.5 Edge Service Client.....	147
5.1.6 ReST API.....	147
5.1.6.1 Login.....	147
5.1.6.1.1 Login request.....	147
5.1.6.1.2 Login response.....	147
5.1.6.1.3 Cookies.....	147
5.1.6.2 Logout.....	147
5.1.6.2.1 Logout request.....	147
5.1.6.2.2 Logout response.....	148
5.1.6.3 Voucher list.....	148
5.1.6.3.1 Voucher list request.....	148
5.1.6.3.2 Voucher list response.....	148
5.1.6.4 Voucher detail.....	148
5.1.6.4.1 Voucher detail request.....	149
5.1.6.4.2 Voucher detail response.....	149
5.1.6.5 Voucher order.....	149
5.1.6.5.1 Voucher order request.....	149
5.1.6.5.2 Voucher order response.....	150
5.1.6.6 Voucher order detail.....	150
5.1.6.6.1 Voucher order detail request.....	151
5.1.6.6.2 Voucher order detail response.....	151
5.1.6.7 Provider Text.....	151
5.1.6.7.1 Provider Text request.....	151
5.1.6.7.2 Provider Text response.....	151
5.1.6.8 Content.....	151
5.1.6.8.1 Content request.....	151
5.1.6.8.2 Content response.....	152
5.1.6.9 Verification.....	152
5.1.6.9.1 Verification request by phone number.....	152
5.1.6.9.2 Verification by phone number response.....	152
5.1.6.9.3 Verification request by sms.....	152
5.1.6.9.4 Verification by sms number response.....	152
5.1.6.10 Cookies.....	152
5.1.6.10.1 Cookies request.....	152
5.1.6.10.2 Cookies response.....	152
5.1.6.11 Documentation.....	152
5.1.6.11.1 Documentation request.....	153
5.1.6.11.2 Documentation response.....	153
5.1.6.12 i18 internationalization.....	153
5.1.6.12.1 i18 request.....	153
5.2 Edge Service.....	153
5.2.1 TODO.....	153



5.2.1.1 TODO version 1.0.....	153
5.2.1.2 TODO version 1.0-next.....	153
5.2.2 Test environment and credentials.....	153
5.2.3 Edge Service Registry.....	153
5.2.4 Edge Service Relay.....	154
5.2.5 Edge Service Client.....	154
5.2.6 ReST API.....	154
5.2.6.1 Login.....	154
5.2.6.1.1 Login request.....	154
5.2.6.1.2 Login response.....	155
5.2.6.1.3 Cookies.....	155
5.2.6.2 Logout.....	155
5.2.6.2.1 Logout request.....	155
5.2.6.2.2 Logout response.....	155
5.2.6.3 Voucher list.....	155
5.2.6.3.1 Voucher list request.....	156
5.2.6.3.2 Voucher list response.....	156
5.2.6.4 Voucher detail.....	156
5.2.6.4.1 Voucher detail request.....	156
5.2.6.4.2 Voucher detail response.....	156
5.2.6.5 Voucher order.....	156
5.2.6.5.1 Voucher order request.....	156
5.2.6.5.2 Voucher order response.....	158
5.2.6.6 Voucher order detail.....	158
5.2.6.6.1 Voucher order detail request.....	158
5.2.6.6.2 Voucher order detail response.....	158
5.2.6.7 Provider Text.....	158
5.2.6.7.1 Provider Text request.....	158
5.2.6.7.2 Provider Text response.....	159
5.2.6.8 Content.....	159
5.2.6.8.1 Content request.....	159
5.2.6.8.2 Content response.....	159
5.2.6.9 Verification.....	159
5.2.6.9.1 Verification request by phone number.....	160
5.2.6.9.2 Verification by phone number response.....	160
5.2.6.9.3 Verification request by sms.....	160
5.2.6.9.4 Verification by sms number response.....	160
5.2.6.10 Cookies.....	160
5.2.6.10.1 Cookies request.....	160
5.2.6.10.2 Cookies response.....	160
5.2.6.11 Documentation.....	160
5.2.6.11.1 Documentation request.....	160
5.2.6.11.2 Documentation response.....	160
5.2.6.12 i18 internationalization.....	161
5.2.6.12.1 i18 request.....	161
5.3 Edge Service.....	161
5.3.1 TODO.....	161
5.3.1.1 TODO version 1.0.....	161
5.3.1.2 TODO version 1.0-next.....	161
5.3.2 Test environment and credentials.....	161
5.3.3 Edge Service Registry.....	161
5.3.4 Edge Service Relay.....	162
5.3.5 Edge Service Client.....	162
5.3.6 ReST API.....	162
5.3.6.1 Login.....	162



5.3.6.1.1 Login request.....	162
5.3.6.1.2 Login response.....	163
5.3.6.1.3 Cookies.....	163
5.3.6.2 Logout.....	163
5.3.6.2.1 Logout request.....	163
5.3.6.2.2 Logout response.....	163
5.3.6.3 Voucher list.....	163
5.3.6.3.1 Voucher list request.....	164
5.3.6.3.2 Voucher list response.....	164
5.3.6.4 Voucher detail.....	164
5.3.6.4.1 Voucher detail request.....	164
5.3.6.4.2 Voucher detail response.....	164
5.3.6.5 Voucher order.....	164
5.3.6.5.1 Voucher order request.....	164
5.3.6.5.2 Voucher order response.....	166
5.3.6.6 Voucher order detail.....	166
5.3.6.6.1 Voucher order detail request.....	166
5.3.6.6.2 Voucher order detail response.....	166
5.3.6.7 Provider Text.....	166
5.3.6.7.1 Provider Text request.....	166
5.3.6.7.2 Provider Text response.....	167
5.3.6.8 Content.....	167
5.3.6.8.1 Content request.....	167
5.3.6.8.2 Content response.....	167
5.3.6.9 Verification.....	167
5.3.6.9.1 Verification request by phone number.....	168
5.3.6.9.2 Verification by phone number response.....	168
5.3.6.9.3 Verification request by sms.....	168
5.3.6.9.4 Verification by sms number response.....	168
5.3.6.10 Cookies.....	168
5.3.6.10.1 Cookies request.....	168
5.3.6.10.2 Cookies response.....	168
5.3.6.11 Documentation.....	168
5.3.6.11.1 Documentation request.....	168
5.3.6.11.2 Documentation response.....	168
5.3.6.12 i18n internationalization.....	169
5.3.6.12.1 i18n request.....	169
5.4 Edge Service.....	169
5.4.1 TODO.....	169
5.4.1.1 TODO version 1.0.....	169
5.4.1.2 TODO version 1.0-next.....	169
5.4.2 Test environment and credentials.....	169
5.4.3 Edge Service Registry.....	169
5.4.4 Edge Service Relay.....	170
5.4.5 Edge Service Client.....	170
5.4.6 ReST API.....	170
5.4.6.1 Login.....	170
5.4.6.1.1 Login request.....	170
5.4.6.1.2 Login response.....	171
5.4.6.1.3 Cookies.....	171
5.4.6.2 Logout.....	171
5.4.6.2.1 Logout request.....	171
5.4.6.2.2 Logout response.....	171
5.4.6.3 Voucher list.....	171
5.4.6.3.1 Voucher list request.....	172



5.4.6.3.2 Voucher list response.....	172
5.4.6.4 Voucher detail.....	172
5.4.6.4.1 Voucher detail request.....	172
5.4.6.4.2 Voucher detail response.....	172
5.4.6.5 Voucher order.....	172
5.4.6.5.1 Voucher order request.....	172
5.4.6.5.2 Voucher order response.....	174
5.4.6.6 Voucher order detail.....	174
5.4.6.6.1 Voucher order detail request.....	174
5.4.6.6.2 Voucher order detail response.....	174
5.4.6.7 Provider Text.....	174
5.4.6.7.1 Provider Text request.....	174
5.4.6.7.2 Provider Text response.....	175
5.4.6.8 Content.....	175
5.4.6.8.1 Content request.....	175
5.4.6.8.2 Content response.....	175
5.4.6.9 Verification.....	175
5.4.6.9.1 Verification request by phone number.....	176
5.4.6.9.2 Verification by phone number response.....	176
5.4.6.9.3 Verification request by sms.....	176
5.4.6.9.4 Verification by sms number response.....	176
5.4.6.10 Cookies.....	176
5.4.6.10.1 Cookies request.....	176
5.4.6.10.2 Cookies response.....	176
5.4.6.11 Documentation.....	176
5.4.6.11.1 Documentation request.....	176
5.4.6.11.2 Documentation response.....	176
5.4.6.12 i18n internationalization.....	177
5.4.6.12.1 i18n request.....	177
5.5 Edge Service.....	177
5.5.1 TODO.....	177
5.5.1.1 TODO version 1.0.....	177
5.5.1.2 TODO version 1.0-next.....	177
5.5.2 Test environment and credentials.....	177
5.5.3 Edge Service Registry.....	177
5.5.4 Edge Service Relay.....	178
5.5.5 Edge Service Client.....	178
5.5.6 ReST API.....	178
5.5.6.1 Login.....	178
5.5.6.1.1 Login request.....	178
5.5.6.1.2 Login response.....	179
5.5.6.1.3 Cookies.....	179
5.5.6.2 Logout.....	179
5.5.6.2.1 Logout request.....	179
5.5.6.2.2 Logout response.....	179
5.5.6.3 Voucher list.....	179
5.5.6.3.1 Voucher list request.....	180
5.5.6.3.2 Voucher list response.....	180
5.5.6.4 Voucher detail.....	180
5.5.6.4.1 Voucher detail request.....	180
5.5.6.4.2 Voucher detail response.....	180
5.5.6.5 Voucher order.....	180
5.5.6.5.1 Voucher order request.....	180
5.5.6.5.2 Voucher order response.....	182
5.5.6.6 Voucher order detail.....	182



5.5.6.6.1 Voucher order detail request.....	182
5.5.6.6.2 Voucher order detail response.....	182
5.5.6.7 Provider Text.....	182
5.5.6.7.1 Provider Text request.....	182
5.5.6.7.2 Provider Text response.....	183
5.5.6.8 Content.....	183
5.5.6.8.1 Content request.....	183
5.5.6.8.2 Content response.....	183
5.5.6.9 Verification.....	183
5.5.6.9.1 Verification request by phone number.....	184
5.5.6.9.2 Verification by phone number response.....	184
5.5.6.9.3 Verification request by sms.....	184
5.5.6.9.4 Verification by sms number response.....	184
5.5.6.10 Cookies.....	184
5.5.6.10.1 Cookies request.....	184
5.5.6.10.2 Cookies response.....	184
5.5.6.11 Documentation.....	184
5.5.6.11.1 Documentation request.....	184
5.5.6.11.2 Documentation response.....	184
5.5.6.12 i18n internationalization.....	185
5.5.6.12.1 i18n request.....	185
6 Big.....	186
6.1 Edge Service.....	186
6.1.1 TODO.....	186
6.1.1.1 TODO version 1.0.....	186
6.1.1.2 TODO version 1.0-next.....	186
6.1.2 Test environment and credentials.....	186
6.1.3 Edge Service Registry.....	186
6.1.4 Edge Service Relay.....	186
6.1.5 Edge Service Client.....	187
6.1.6 ReST API.....	187
6.1.6.1 Login.....	187
6.1.6.1.1 Login request.....	187
6.1.6.1.2 Login response.....	187
6.1.6.1.3 Cookies.....	187
6.1.6.2 Logout.....	187
6.1.6.2.1 Logout request.....	187
6.1.6.2.2 Logout response.....	188
6.1.6.3 Voucher list.....	188
6.1.6.3.1 Voucher list request.....	188
6.1.6.3.2 Voucher list response.....	188
6.1.6.4 Voucher detail.....	188
6.1.6.4.1 Voucher detail request.....	189
6.1.6.4.2 Voucher detail response.....	189
6.1.6.5 Voucher order.....	189
6.1.6.5.1 Voucher order request.....	189
6.1.6.5.2 Voucher order response.....	190
6.1.6.6 Voucher order detail.....	190
6.1.6.6.1 Voucher order detail request.....	191
6.1.6.6.2 Voucher order detail response.....	191
6.1.6.7 Provider Text.....	191
6.1.6.7.1 Provider Text request.....	191
6.1.6.7.2 Provider Text response.....	191
6.1.6.8 Content.....	191



6.1.6.8.1 Content request.....	191
6.1.6.8.2 Content response.....	192
6.1.6.9 Verification.....	192
6.1.6.9.1 Verification request by phone number.....	192
6.1.6.9.2 Verification by phone number response.....	192
6.1.6.9.3 Verification request by sms.....	192
6.1.6.9.4 Verification by sms number response.....	192
6.1.6.10 Cookies.....	192
6.1.6.10.1 Cookies request.....	192
6.1.6.10.2 Cookies response.....	192
6.1.6.11 Documentation.....	192
6.1.6.11.1 Documentation request.....	193
6.1.6.11.2 Documentation response.....	193
6.1.6.12 i18n internationalization.....	193
6.1.6.12.1 i18n request.....	193
6.2 Edge Service.....	193
6.2.1 TODO.....	193
6.2.1.1 TODO version 1.0.....	193
6.2.1.2 TODO version 1.0-next.....	193
6.2.2 Test environment and credentials.....	193
6.2.3 Edge Service Registry.....	193
6.2.4 Edge Service Relay.....	194
6.2.5 Edge Service Client.....	194
6.2.6 ReST API.....	194
6.2.6.1 Login.....	194
6.2.6.1.1 Login request.....	194
6.2.6.1.2 Login response.....	195
6.2.6.1.3 Cookies.....	195
6.2.6.2 Logout.....	195
6.2.6.2.1 Logout request.....	195
6.2.6.2.2 Logout response.....	195
6.2.6.3 Voucher list.....	195
6.2.6.3.1 Voucher list request.....	196
6.2.6.3.2 Voucher list response.....	196
6.2.6.4 Voucher detail.....	196
6.2.6.4.1 Voucher detail request.....	196
6.2.6.4.2 Voucher detail response.....	196
6.2.6.5 Voucher order.....	196
6.2.6.5.1 Voucher order request.....	196
6.2.6.5.2 Voucher order response.....	198
6.2.6.6 Voucher order detail.....	198
6.2.6.6.1 Voucher order detail request.....	198
6.2.6.6.2 Voucher order detail response.....	198
6.2.6.7 Provider Text.....	198
6.2.6.7.1 Provider Text request.....	198
6.2.6.7.2 Provider Text response.....	199
6.2.6.8 Content.....	199
6.2.6.8.1 Content request.....	199
6.2.6.8.2 Content response.....	199
6.2.6.9 Verification.....	199
6.2.6.9.1 Verification request by phone number.....	200
6.2.6.9.2 Verification by phone number response.....	200
6.2.6.9.3 Verification request by sms.....	200
6.2.6.9.4 Verification by sms number response.....	200
6.2.6.10 Cookies.....	200



6.2.6.10.1 Cookies request.....	200
6.2.6.10.2 Cookies response.....	200
6.2.6.11 Documentation.....	200
6.2.6.11.1 Documentation request.....	200
6.2.6.11.2 Documentation response.....	200
6.2.6.12 i18 internationalization.....	201
6.2.6.12.1 i18 request.....	201
6.3 Edge Service.....	201
6.3.1 TODO.....	201
6.3.1.1 TODO version 1.0.....	201
6.3.1.2 TODO version 1.0-next.....	201
6.3.2 Test environment and credentials.....	201
6.3.3 Edge Service Registry.....	201
6.3.4 Edge Service Relay.....	202
6.3.5 Edge Service Client.....	202
6.3.6 ReST API.....	202
6.3.6.1 Login.....	202
6.3.6.1.1 Login request.....	202
6.3.6.1.2 Login response.....	203
6.3.6.1.3 Cookies.....	203
6.3.6.2 Logout.....	203
6.3.6.2.1 Logout request.....	203
6.3.6.2.2 Logout response.....	203
6.3.6.3 Voucher list.....	203
6.3.6.3.1 Voucher list request.....	204
6.3.6.3.2 Voucher list response.....	204
6.3.6.4 Voucher detail.....	204
6.3.6.4.1 Voucher detail request.....	204
6.3.6.4.2 Voucher detail response.....	204
6.3.6.5 Voucher order.....	204
6.3.6.5.1 Voucher order request.....	204
6.3.6.5.2 Voucher order response.....	206
6.3.6.6 Voucher order detail.....	206
6.3.6.6.1 Voucher order detail request.....	206
6.3.6.6.2 Voucher order detail response.....	206
6.3.6.7 Provider Text.....	206
6.3.6.7.1 Provider Text request.....	206
6.3.6.7.2 Provider Text response.....	207
6.3.6.8 Content.....	207
6.3.6.8.1 Content request.....	207
6.3.6.8.2 Content response.....	207
6.3.6.9 Verification.....	207
6.3.6.9.1 Verification request by phone number.....	208
6.3.6.9.2 Verification by phone number response.....	208
6.3.6.9.3 Verification request by sms.....	208
6.3.6.9.4 Verification by sms number response.....	208
6.3.6.10 Cookies.....	208
6.3.6.10.1 Cookies request.....	208
6.3.6.10.2 Cookies response.....	208
6.3.6.11 Documentation.....	208
6.3.6.11.1 Documentation request.....	208
6.3.6.11.2 Documentation response.....	208
6.3.6.12 i18 internationalization.....	209
6.3.6.12.1 i18 request.....	209



6.4 Edge Service.....	209
6.4.1 TODO.....	209
6.4.1.1 TODO version 1.0.....	209
6.4.1.2 TODO version 1.0-next.....	209
6.4.2 Test environment and credentials.....	209
6.4.3 Edge Service Registry.....	209
6.4.4 Edge Service Relay.....	210
6.4.5 Edge Service Client.....	210
6.4.6 ReST API.....	210
6.4.6.1 Login.....	210
6.4.6.1.1 Login request.....	210
6.4.6.1.2 Login response.....	211
6.4.6.1.3 Cookies.....	211
6.4.6.2 Logout.....	211
6.4.6.2.1 Logout request.....	211
6.4.6.2.2 Logout response.....	211
6.4.6.3 Voucher list.....	211
6.4.6.3.1 Voucher list request.....	212
6.4.6.3.2 Voucher list response.....	212
6.4.6.4 Voucher detail.....	212
6.4.6.4.1 Voucher detail request.....	212
6.4.6.4.2 Voucher detail response.....	212
6.4.6.5 Voucher order.....	212
6.4.6.5.1 Voucher order request.....	212
6.4.6.5.2 Voucher order response.....	214
6.4.6.6 Voucher order detail.....	214
6.4.6.6.1 Voucher order detail request.....	214
6.4.6.6.2 Voucher order detail response.....	214
6.4.6.7 Provider Text.....	214
6.4.6.7.1 Provider Text request.....	214
6.4.6.7.2 Provider Text response.....	215
6.4.6.8 Content.....	215
6.4.6.8.1 Content request.....	215
6.4.6.8.2 Content response.....	215
6.4.6.9 Verification.....	215
6.4.6.9.1 Verification request by phone number.....	216
6.4.6.9.2 Verification by phone number response.....	216
6.4.6.9.3 Verification request by sms.....	216
6.4.6.9.4 Verification by sms number response.....	216
6.4.6.10 Cookies.....	216
6.4.6.10.1 Cookies request.....	216
6.4.6.10.2 Cookies response.....	216
6.4.6.11 Documentation.....	216
6.4.6.11.1 Documentation request.....	216
6.4.6.11.2 Documentation response.....	216
6.4.6.12 i18 internationalization.....	217
6.4.6.12.1 i18 request.....	217
6.5 Edge Service.....	217
6.5.1 TODO.....	217
6.5.1.1 TODO version 1.0.....	217
6.5.1.2 TODO version 1.0-next.....	217
6.5.2 Test environment and credentials.....	217
6.5.3 Edge Service Registry.....	217
6.5.4 Edge Service Relay.....	218
6.5.5 Edge Service Client.....	218



6.5.6 ReST API.....	218
6.5.6.1 Login.....	218
6.5.6.1.1 Login request.....	218
6.5.6.1.2 Login response.....	219
6.5.6.1.3 Cookies.....	219
6.5.6.2 Logout.....	219
6.5.6.2.1 Logout request.....	219
6.5.6.2.2 Logout response.....	219
6.5.6.3 Voucher list.....	219
6.5.6.3.1 Voucher list request.....	220
6.5.6.3.2 Voucher list response.....	220
6.5.6.4 Voucher detail.....	220
6.5.6.4.1 Voucher detail request.....	220
6.5.6.4.2 Voucher detail response.....	220
6.5.6.5 Voucher order.....	220
6.5.6.5.1 Voucher order request.....	220
6.5.6.5.2 Voucher order response.....	222
6.5.6.6 Voucher order detail.....	222
6.5.6.6.1 Voucher order detail request.....	222
6.5.6.6.2 Voucher order detail response.....	222
6.5.6.7 Provider Text.....	222
6.5.6.7.1 Provider Text request.....	222
6.5.6.7.2 Provider Text response.....	223
6.5.6.8 Content.....	223
6.5.6.8.1 Content request.....	223
6.5.6.8.2 Content response.....	223
6.5.6.9 Verification.....	223
6.5.6.9.1 Verification request by phone number.....	224
6.5.6.9.2 Verification by phone number response.....	224
6.5.6.9.3 Verification request by sms.....	224
6.5.6.9.4 Verification by sms number response.....	224
6.5.6.10 Cookies.....	224
6.5.6.10.1 Cookies request.....	224
6.5.6.10.2 Cookies response.....	224
6.5.6.11 Documentation.....	224
6.5.6.11.1 Documentation request.....	224
6.5.6.11.2 Documentation response.....	224
6.5.6.12 i18n internationalization.....	225
6.5.6.12.1 i18n request.....	225

Table of Tables

Table 1: Document references.....	25
-----------------------------------	----

Table of Figures



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 Big

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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI
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	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	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"



field name	type	constrains	description	example
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-h0mLD2zDgf26ptjxSNRiKFILM",      "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"    }  } }
```

2.1.6.6 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-h0mLD2zDgf26ptjxSNRiKFILM",						
"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 [GetProviderTexts] 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						

2.2 Edge Service

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

2.2.1 TODO

2.2.1.1 TODO version 1.0

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

2.2.1.2 TODO version 1.0-next

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

2.2.2 Test environment and credentials

TODO

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

2.2.3 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. Therefore 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.

2.2.4 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.

2.2.5 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.

2.2.6 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].

2.2.6.1 Login

2.2.6.1.1 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}	MzAwMDAwMTY6dG9wLXNlY3Jl dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

2.2.6.1.2 Login response

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

2.2.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

2.2.6.2 Logout

2.2.6.2.1 Logout request

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

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

2.2.6.2.2 Logout response

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

2.2.6.3 Voucher list

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



2.2.6.3.1 Voucher list request

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

2.2.6.3.2 Voucher list response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": [
    {
      "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": "f634f09f-9a06-4297-a326-0d703938b702"
    },
    {
      "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"
          }
        }
      },
      "id": "a3473e20-19b4-44fe-8389-67a511854759"
    }
  ]
}
```

2.2.6.4 Voucher detail

2.2.6.4.1 Voucher detail request

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

2.2.6.4.2 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"
        }
      }
    }
  }
}
```

2.2.6.5 Voucher order

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

2.2.6.5.1 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



2.2.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

2.2.6.6 Voucher order detail

2.2.6.6.1 Voucher order detail request

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

2.2.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

2.2.6.7 Provider Text

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

2.2.6.7.1 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
```

2.2.6.7.2 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."    }    }  ] }
```

2.2.6.8 Content

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

2.2.6.8.1 Content request

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

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

2.2.6.8.2 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",    }  } }
```

2.2.6.9 Verification

See for more information on the source of these fields.



2.2.6.9.1 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" } } }
```

2.2.6.9.2 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 } } }
```

2.2.6.9.3 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" } } }
```

2.2.6.9.4 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 } } }
```

2.2.6.10 Cookies

2.2.6.10.1 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" } } ] }
```

2.2.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

2.2.6.11 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.

2.2.6.11.1 Documentation request

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

2.2.6.11.2 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"
  ]
}

```

2.2.6.12 i18 internationalization

- ✓ Retrieves i18next json based on requested language

2.2.6.12.1 i18 request

```

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

```

2.3 Edge Service

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

2.3.1 TODO

2.3.1.1 TODO version 1.0

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

2.3.1.2 TODO version 1.0-next

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

2.3.2 Test environment and credentials

TODO

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

2.3.3 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.

2.3.4 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.](#)

2.3.5 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.

2.3.6 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].

2.3.6.1 Login

2.3.6.1.1 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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

2.3.6.1.2 Login response

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

2.3.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

2.3.6.2 Logout

2.3.6.2.1 Logout request

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

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

2.3.6.2.2 Logout response

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

2.3.6.3 Voucher list

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



2.3.6.3.1 Voucher list request

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

2.3.6.3.2 Voucher list response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": [
    {
      "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": "f634f09f-9a06-4297-a326-0d703938b702"
    },
    {
      "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"
          }
        }
      },
      "id": "a3473e20-19b4-44fe-8389-67a511854759"
    }
  ]
}
```

2.3.6.4 Voucher detail

2.3.6.4.1 Voucher detail request

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

2.3.6.4.2 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"
        }
      }
    }
  }
}
```

2.3.6.5 Voucher order

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

2.3.6.5.1 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



2.3.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

2.3.6.6 Voucher order detail

2.3.6.6.1 Voucher order detail request

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

2.3.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

2.3.6.7 Provider Text

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

2.3.6.7.1 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
```

2.3.6.7.2 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."      }    }  ] }
```

2.3.6.8 Content

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

2.3.6.8.1 Content request

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

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

2.3.6.8.2 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",    }  } }
```

2.3.6.9 Verification

See for more information on the source of these fields.



2.3.6.9.1 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" } } }
```

2.3.6.9.2 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 } } }
```

2.3.6.9.3 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" } } }
```

2.3.6.9.4 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 } } }
```

2.3.6.10 Cookies

2.3.6.10.1 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" } } ] }
```

2.3.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

2.3.6.11 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.

2.3.6.11.1 Documentation request

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

2.3.6.11.2 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"
  ]
}

```

2.3.6.12 *i18 internationalization*

- ✓ *Retrieves i18next json based on requested language*

2.3.6.12.1 *i18 request*

```

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

```

2.4 Edge Service

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

2.4.1 *TODO*

2.4.1.1 *TODO version 1.0*

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

2.4.1.2 *TODO version 1.0-next*

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

2.4.2 *Test environment and credentials*

TODO

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

2.4.3 *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. Therefore 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.

2.4.4 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.](#)

2.4.5 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.

2.4.6 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].

2.4.6.1 Login

2.4.6.1.1 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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

2.4.6.1.2 Login response

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

2.4.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

2.4.6.2 Logout

2.4.6.2.1 Logout request

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

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

2.4.6.2.2 Logout response

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

2.4.6.3 Voucher list

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



2.4.6.3.1 Voucher list request

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

2.4.6.3.2 Voucher list response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": [
    {
      "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": "f634f09f-9a06-4297-a326-0d703938b702"
    },
    {
      "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"
          }
        }
      },
      "id": "a3473e20-19b4-44fe-8389-67a511854759"
    }
  ]
}
```

2.4.6.4 Voucher detail

2.4.6.4.1 Voucher detail request

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

2.4.6.4.2 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"
        }
      }
    }
  }
}
```

2.4.6.5 Voucher order

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

2.4.6.5.1 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



2.4.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

2.4.6.6 Voucher order detail

2.4.6.6.1 Voucher order detail request

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

2.4.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

2.4.6.7 Provider Text

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

2.4.6.7.1 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
```

2.4.6.7.2 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."      }    }  ] }
```

2.4.6.8 Content

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

2.4.6.8.1 Content request

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

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

2.4.6.8.2 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",    }  } }
```

2.4.6.9 Verification

See for more information on the source of these fields.



2.4.6.9.1 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" } } }
```

2.4.6.9.2 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 } } }
```

2.4.6.9.3 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" } } }
```

2.4.6.9.4 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 } } }
```

2.4.6.10 Cookies

2.4.6.10.1 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" } } ] }
```

2.4.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

2.4.6.11 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.

2.4.6.11.1 Documentation request

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

2.4.6.11.2 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"
] }

```

2.4.6.12 i18 internationalization

- ✓ Retrieves i18next json based on requested language

2.4.6.12.1 i18 request

```

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

```

2.5 Edge Service

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

2.5.1 TODO

2.5.1.1 TODO version 1.0

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

2.5.1.2 TODO version 1.0-next

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

2.5.2 Test environment and credentials

TODO

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

2.5.3 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. Therefore 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 unregister on shutdown.

2.5.4 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.](#)

2.5.5 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.

2.5.6 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].

2.5.6.1 Login

2.5.6.1.1 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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

2.5.6.1.2 Login response

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

2.5.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

2.5.6.2 Logout

2.5.6.2.1 Logout request

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

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

2.5.6.2.2 Logout response

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

2.5.6.3 Voucher list

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



2.5.6.3.1 Voucher list request

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

2.5.6.3.2 Voucher list response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": [
    {
      "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": "f634f09f-9a06-4297-a326-0d703938b702"
    },
    {
      "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"
          }
        }
      },
      "id": "a3473e20-19b4-44fe-8389-67a511854759"
    }
  ]
}
```

2.5.6.4 Voucher detail

2.5.6.4.1 Voucher detail request

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

2.5.6.4.2 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"
        }
      }
    }
  }
}
```

2.5.6.5 Voucher order

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

2.5.6.5.1 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



2.5.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

2.5.6.6 Voucher order detail

2.5.6.6.1 Voucher order detail request

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

2.5.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

2.5.6.7 Provider Text

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

2.5.6.7.1 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
```

2.5.6.7.2 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."    }    }  ] }
```

2.5.6.8 Content

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

2.5.6.8.1 Content request

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

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

2.5.6.8.2 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",    }    }  }
```

2.5.6.9 Verification

See for more information on the source of these fields.



2.5.6.9.1 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" } } }
```

2.5.6.9.2 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 } } }
```

2.5.6.9.3 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" } } }
```

2.5.6.9.4 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 } } }
```

2.5.6.10 Cookies

2.5.6.10.1 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" } } ] }
```

2.5.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

2.5.6.11 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.

2.5.6.11.1 Documentation request

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

2.5.6.11.2 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"
] }
```

2.5.6.12 i18 internationalization

✓ *Retrieves i18next json based on requested language*

2.5.6.12.1 i18 request

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



3 Big

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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI
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	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	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"



field name	type	constrains	description	example
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

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-h0mLD2zDgf26ptjxSNRiKFILM",      "payment-session-id": "4b7c0aec-3db3-4cd2-9b22-4d41fb964fdc",      "payment-status": "pending",      "payment-method": "creditcard",      "voucher-code": "LEBALLINN10",      "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"    }  } }
```

3.1.6.6 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-h0mLD2zDgf26ptjxSNRiKFILM", "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 [GetProviderTexts] 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						

3.2 Edge Service

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

3.2.1 TODO

3.2.1.1 TODO version 1.0

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

3.2.1.2 TODO version 1.0-next

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

3.2.2 Test environment and credentials

TODO

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

3.2.3 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. Therefore 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.

3.2.4 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.

3.2.5 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.

3.2.6 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].

3.2.6.1 Login

3.2.6.1.1 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}	MzAwMDAwMTY6dG9wLXNlY3Jl dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

3.2.6.1.2 Login response

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

3.2.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

3.2.6.2 Logout

3.2.6.2.1 Logout request

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

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

3.2.6.2.2 Logout response

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

3.2.6.3 Voucher list

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



3.2.6.3.1 Voucher list request

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

3.2.6.3.2 Voucher list response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": [
    {
      "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": "f634f09f-9a06-4297-a326-0d703938b702"
    },
    {
      "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"
          }
        }
      },
      "id": "a3473e20-19b4-44fe-8389-67a511854759"
    }
  ]
}
```

3.2.6.4 Voucher detail

3.2.6.4.1 Voucher detail request

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

3.2.6.4.2 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"
        }
      }
    }
  }
}
```

3.2.6.5 Voucher order

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

3.2.6.5.1 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



3.2.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

3.2.6.6 Voucher order detail

3.2.6.6.1 Voucher order detail request

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

3.2.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

3.2.6.7 Provider Text

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

3.2.6.7.1 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
```

3.2.6.7.2 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."      }    }  ] }
```

3.2.6.8 Content

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

3.2.6.8.1 Content request

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

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

3.2.6.8.2 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",    }  } }
```

3.2.6.9 Verification

See for more information on the source of these fields.



3.2.6.9.1 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" } } }
```

3.2.6.9.2 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 } } }
```

3.2.6.9.3 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" } } }
```

3.2.6.9.4 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 } } }
```

3.2.6.10 Cookies

3.2.6.10.1 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" } } ] }
```

3.2.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

3.2.6.11 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.

3.2.6.11.1 Documentation request

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

3.2.6.11.2 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"
] }
```

3.2.6.12 i18 internationalization

- ✓ Retrieves i18next json based on requested language

3.2.6.12.1 i18 request

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

3.3 Edge Service

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

3.3.1 TODO

3.3.1.1 TODO version 1.0

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

3.3.1.2 TODO version 1.0-next

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

3.3.2 Test environment and credentials

TODO

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

3.3.3 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. Therefore 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 unregister on shutdown.

3.3.4 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.](#)

3.3.5 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.

3.3.6 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].

3.3.6.1 Login

3.3.6.1.1 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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

3.3.6.1.2 Login response

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

3.3.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

3.3.6.2 Logout

3.3.6.2.1 Logout request

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

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

3.3.6.2.2 Logout response

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

3.3.6.3 Voucher list

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



3.3.6.3.1 Voucher list request

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

3.3.6.3.2 Voucher list response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": [
    {
      "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": "f634f09f-9a06-4297-a326-0d703938b702"
    },
    {
      "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"
          }
        }
      },
      "id": "a3473e20-19b4-44fe-8389-67a511854759"
    }
  ]
}
```

3.3.6.4 Voucher detail

3.3.6.4.1 Voucher detail request

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

3.3.6.4.2 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"
        }
      }
    }
  }
}
```

3.3.6.5 Voucher order

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

3.3.6.5.1 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



3.3.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"    }  } }
```

3.3.6.6 Voucher order detail

3.3.6.6.1 Voucher order detail request

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

3.3.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"    }  } }
```

3.3.6.7 Provider Text

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

3.3.6.7.1 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
```

3.3.6.7.2 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."    }    }  ] }
```

3.3.6.8 Content

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

3.3.6.8.1 Content request

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

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

3.3.6.8.2 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",    }  } }
```

3.3.6.9 Verification

See for more information on the source of these fields.



3.3.6.9.1 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" } } }
```

3.3.6.9.2 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 } } }
```

3.3.6.9.3 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" } } }
```

3.3.6.9.4 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 } } }
```

3.3.6.10 Cookies

3.3.6.10.1 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" } } ] }
```

3.3.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

3.3.6.11 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.

3.3.6.11.1 Documentation request

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

3.3.6.11.2 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"
] }
```

3.3.6.12 i18 internationalization

- ✓ *Retrieves i18next json based on requested language*

3.3.6.12.1 i18 request

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

3.4 Edge Service

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

3.4.1 TODO

3.4.1.1 TODO version 1.0

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

3.4.1.2 TODO version 1.0-next

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

3.4.2 Test environment and credentials

TODO

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

3.4.3 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. Therefore 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 unregister on shutdown.

3.4.4 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.](#)

3.4.5 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.

3.4.6 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].

3.4.6.1 Login

3.4.6.1.1 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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

3.4.6.1.2 Login response

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

3.4.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

3.4.6.2 Logout

3.4.6.2.1 Logout request

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

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

3.4.6.2.2 Logout response

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

3.4.6.3 Voucher list

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



3.4.6.3.1 Voucher list request

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

3.4.6.3.2 Voucher list response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": [
    {
      "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": "f634f09f-9a06-4297-a326-0d703938b702"
    },
    {
      "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"
          }
        }
      },
      "id": "a3473e20-19b4-44fe-8389-67a511854759"
    }
  ]
}
```

3.4.6.4 Voucher detail

3.4.6.4.1 Voucher detail request

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

3.4.6.4.2 Voucher detail response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": {
    "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": "f634f09f-9a06-4297-a326-0d703938b702"
  }
}
```

3.4.6.5 Voucher order

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

3.4.6.5.1 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



3.4.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

3.4.6.6 Voucher order detail

3.4.6.6.1 Voucher order detail request

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

3.4.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

3.4.6.7 Provider Text

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

3.4.6.7.1 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
```

3.4.6.7.2 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."      }    }  ] }
```

3.4.6.8 Content

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

3.4.6.8.1 Content request

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

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

3.4.6.8.2 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",    }  } }
```

3.4.6.9 Verification

See for more information on the source of these fields.



3.4.6.9.1 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" } } }
```

3.4.6.9.2 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 } } }
```

3.4.6.9.3 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" } } }
```

3.4.6.9.4 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 } } }
```

3.4.6.10 Cookies

3.4.6.10.1 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" } } ] }
```

3.4.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

3.4.6.11 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.

3.4.6.11.1 Documentation request

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

3.4.6.11.2 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"
] }
```

3.4.6.12 i18 internationalization

- ✓ Retrieves i18next json based on requested language

3.4.6.12.1 i18 request

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

3.5 Edge Service

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

3.5.1 TODO

3.5.1.1 TODO version 1.0

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

3.5.1.2 TODO version 1.0-next

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

3.5.2 Test environment and credentials

TODO

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

3.5.3 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. Therefore 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 unregister on shutdown.

3.5.4 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.](#)

3.5.5 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.

3.5.6 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].

3.5.6.1 Login

3.5.6.1.1 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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

3.5.6.1.2 Login response

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

3.5.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

3.5.6.2 Logout

3.5.6.2.1 Logout request

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

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

3.5.6.2.2 Logout response

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

3.5.6.3 Voucher list

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



3.5.6.3.1 Voucher list request

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

3.5.6.3.2 Voucher list response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": [
    {
      "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": "f634f09f-9a06-4297-a326-0d703938b702"
    },
    {
      "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"
          }
        }
      },
      "id": "a3473e20-19b4-44fe-8389-67a511854759"
    }
  ]
}
```

3.5.6.4 Voucher detail

3.5.6.4.1 Voucher detail request

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

3.5.6.4.2 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"
        }
      }
    }
  }
}
```

3.5.6.5 Voucher order

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

3.5.6.5.1 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



3.5.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

3.5.6.6 Voucher order detail

3.5.6.6.1 Voucher order detail request

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

3.5.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

3.5.6.7 Provider Text

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

3.5.6.7.1 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
```

3.5.6.7.2 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."    }    }  ] }
```

3.5.6.8 Content

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

3.5.6.8.1 Content request

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

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

3.5.6.8.2 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",    }  } }
```

3.5.6.9 Verification

See for more information on the source of these fields.



3.5.6.9.1 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" } } }
```

3.5.6.9.2 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 } } }
```

3.5.6.9.3 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" } } }
```

3.5.6.9.4 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 } } }
```

3.5.6.10 Cookies

3.5.6.10.1 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" } } ] }
```

3.5.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

3.5.6.11 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.

3.5.6.11.1 Documentation request

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

3.5.6.11.2 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"
] }
```

3.5.6.12 i18 internationalization

✓ *Retrieves i18next json based on requested language*

3.5.6.12.1 i18 request

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



4 Big

4.1 Edge Service

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

4.1.1 TODO

4.1.1.1 TODO version 1.0

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

4.1.1.2 TODO version 1.0-next

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

4.1.2 Test environment and credentials

TODO

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

4.1.3 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.

4.1.4 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.](#)

4.1.5 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.

4.1.6 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].

4.1.6.1 Login

4.1.6.1.1 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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic \${ICP_SHOP_DIGEST} Accept: application/vnd.api+json		

4.1.6.1.2 Login response

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

4.1.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Session	true	false

4.1.6.2 Logout

4.1.6.2.1 Logout request

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



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

4.1.6.2.2 Logout response

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

4.1.6.3 Voucher list

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

4.1.6.3.1 Voucher list request

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

4.1.6.3.2 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"
              }
            }
          ]
        }
      }
    }
  ]
}
```

4.1.6.4 Voucher detail



4.1.6.4.1 Voucher detail request

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

4.1.6.4.2 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"
        }
      }
    }
  }
}
```

4.1.6.5 Voucher order

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

4.1.6.5.1 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"



field name	type	constrains	description	example
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

4.1.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFILM",      "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"    }  } }
```

4.1.6.6 Voucher order detail



4.1.6.6.1 Voucher order detail request

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

4.1.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFILM",      "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"    }  } }
```

4.1.6.7 Provider Text

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

4.1.6.7.1 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
```

4.1.6.7.2 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."      }    }  ] }
```

4.1.6.8 Content

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

4.1.6.8.1 Content request

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



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

4.1.6.8.2 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",  } }
```

4.1.6.9 Verification

See for more information on the source of these fields.

4.1.6.9.1 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"    }  } }
```

4.1.6.9.2 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    }  } }
```

4.1.6.9.3 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"    }  } }
```

4.1.6.9.4 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    }  } }
```

4.1.6.10 Cookies

4.1.6.10.1 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"      }    }  ] }
```

4.1.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

4.1.6.11 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.



4.1.6.11.1 Documentation request

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

4.1.6.11.2 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"
] }

4.1.6.12 i18 internationalization

- ✓ Retrieves i18next json based on requested language

4.1.6.12.1 i18 request

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

4.2 Edge Service

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

4.2.1 TODO

4.2.1.1 TODO version 1.0

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

4.2.1.2 TODO version 1.0-next

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

4.2.2 Test environment and credentials

TODO

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

4.2.3 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. Therefore 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.

4.2.4 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.

4.2.5 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.

4.2.6 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].

4.2.6.1 Login

4.2.6.1.1 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}	MzAwMDAwMTY6dG9wLXNlY3Jl dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

4.2.6.1.2 Login response

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

4.2.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

4.2.6.2 Logout

4.2.6.2.1 Logout request

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

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

4.2.6.2.2 Logout response

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

4.2.6.3 Voucher list

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



4.2.6.3.1 Voucher list request

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

4.2.6.3.2 Voucher list response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": [
    {
      "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": "f634f09f-9a06-4297-a326-0d703938b702"
    },
    {
      "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"
          }
        }
      },
      "id": "a3473e20-19b4-44fe-8389-67a511854759"
    }
  ]
}
```

4.2.6.4 Voucher detail

4.2.6.4.1 Voucher detail request

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

4.2.6.4.2 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"
        }
      }
    }
  }
}
```

4.2.6.5 Voucher order

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

4.2.6.5.1 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



4.2.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

4.2.6.6 Voucher order detail

4.2.6.6.1 Voucher order detail request

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

4.2.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

4.2.6.7 Provider Text

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

4.2.6.7.1 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
```

4.2.6.7.2 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."      }    }  ] }
```

4.2.6.8 Content

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

4.2.6.8.1 Content request

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

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

4.2.6.8.2 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",    }  } }
```

4.2.6.9 Verification

See for more information on the source of these fields.



4.2.6.9.1 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" } } }
```

4.2.6.9.2 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 } } }
```

4.2.6.9.3 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" } } }
```

4.2.6.9.4 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 } } }
```

4.2.6.10 Cookies

4.2.6.10.1 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" } } ] }
```

4.2.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

4.2.6.11 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.

4.2.6.11.1 Documentation request

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

4.2.6.11.2 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"
] }
```

4.2.6.12 i18 internationalization

- ✓ Retrieves i18next json based on requested language

4.2.6.12.1 i18 request

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

4.3 Edge Service

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

4.3.1 TODO

4.3.1.1 TODO version 1.0

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

4.3.1.2 TODO version 1.0-next

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

4.3.2 Test environment and credentials

TODO

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

4.3.3 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. Therefore 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 unregister on shutdown.

4.3.4 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.](#)

4.3.5 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.

4.3.6 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].

4.3.6.1 Login

4.3.6.1.1 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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

4.3.6.1.2 Login response

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

4.3.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

4.3.6.2 Logout

4.3.6.2.1 Logout request

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

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

4.3.6.2.2 Logout response

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

4.3.6.3 Voucher list

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



4.3.6.3.1 Voucher list request

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

4.3.6.3.2 Voucher list response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": [
    {
      "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": "f634f09f-9a06-4297-a326-0d703938b702"
    },
    {
      "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"
          }
        }
      },
      "id": "a3473e20-19b4-44fe-8389-67a511854759"
    }
  ]
}
```

4.3.6.4 Voucher detail

4.3.6.4.1 Voucher detail request

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

4.3.6.4.2 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"
        }
      }
    }
  }
}
```

4.3.6.5 Voucher order

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

4.3.6.5.1 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



4.3.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

4.3.6.6 Voucher order detail

4.3.6.6.1 Voucher order detail request

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

4.3.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

4.3.6.7 Provider Text

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

4.3.6.7.1 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
```

4.3.6.7.2 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."      }    }  ] }
```

4.3.6.8 Content

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

4.3.6.8.1 Content request

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

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

4.3.6.8.2 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",    }  } }
```

4.3.6.9 Verification

See for more information on the source of these fields.



4.3.6.9.1 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" } } }
```

4.3.6.9.2 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 } } }
```

4.3.6.9.3 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" } } }
```

4.3.6.9.4 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 } } }
```

4.3.6.10 Cookies

4.3.6.10.1 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" } } ] }
```

4.3.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

4.3.6.11 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.

4.3.6.11.1 Documentation request

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

4.3.6.11.2 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"
  ]
}

```

4.3.6.12 *i18 internationalization*

- ✓ *Retrieves i18next json based on requested language*

4.3.6.12.1 *i18 request*

```

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

```

4.4 Edge Service

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

4.4.1 *TODO*

4.4.1.1 *TODO version 1.0*

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

4.4.1.2 *TODO version 1.0-next*

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

4.4.2 *Test environment and credentials*

TODO

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

4.4.3 *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.

4.4.4 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.](#)

4.4.5 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.

4.4.6 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].

4.4.6.1 Login

4.4.6.1.1 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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

4.4.6.1.2 Login response

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

4.4.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

4.4.6.2 Logout

4.4.6.2.1 Logout request

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

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

4.4.6.2.2 Logout response

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

4.4.6.3 Voucher list

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



4.4.6.3.1 Voucher list request

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

4.4.6.3.2 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"
          }
        }
      }
    }
  ]
}
```

4.4.6.4 Voucher detail

4.4.6.4.1 Voucher detail request

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

4.4.6.4.2 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"
        }
      }
    }
  }
}
```

4.4.6.5 Voucher order

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

4.4.6.5.1 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



4.4.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

4.4.6.6 Voucher order detail

4.4.6.6.1 Voucher order detail request

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

4.4.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

4.4.6.7 Provider Text

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

4.4.6.7.1 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
```

4.4.6.7.2 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."      }    }  ] }
```

4.4.6.8 Content

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

4.4.6.8.1 Content request

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

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

4.4.6.8.2 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",    }  } }
```

4.4.6.9 Verification

See for more information on the source of these fields.



4.4.6.9.1 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" } } }
```

4.4.6.9.2 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 } } }
```

4.4.6.9.3 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" } } }
```

4.4.6.9.4 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 } } }
```

4.4.6.10 Cookies

4.4.6.10.1 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" } } ] }
```

4.4.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

4.4.6.11 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.

4.4.6.11.1 Documentation request

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

4.4.6.11.2 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"
] }

```

4.4.6.12 i18 internationalization

- ✓ Retrieves i18next json based on requested language

4.4.6.12.1 i18 request

```

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

```

4.5 Edge Service

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

4.5.1 TODO

4.5.1.1 TODO version 1.0

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

4.5.1.2 TODO version 1.0-next

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

4.5.2 Test environment and credentials

TODO

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

4.5.3 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.

4.5.4 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.](#)

4.5.5 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.

4.5.6 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].

4.5.6.1 Login

4.5.6.1.1 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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

4.5.6.1.2 Login response

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

4.5.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

4.5.6.2 Logout

4.5.6.2.1 Logout request

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

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

4.5.6.2.2 Logout response

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

4.5.6.3 Voucher list

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



4.5.6.3.1 Voucher list request

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

4.5.6.3.2 Voucher list response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": [
    {
      "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": "f634f09f-9a06-4297-a326-0d703938b702"
    },
    {
      "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"
          }
        }
      },
      "id": "a3473e20-19b4-44fe-8389-67a511854759"
    }
  ]
}
```

4.5.6.4 Voucher detail

4.5.6.4.1 Voucher detail request

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

4.5.6.4.2 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"
        }
      }
    }
  }
}
```

4.5.6.5 Voucher order

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

4.5.6.5.1 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



4.5.6.5.2 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"      }      } }
```

4.5.6.6 Voucher order detail

4.5.6.6.1 Voucher order detail request

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

4.5.6.6.2 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"      }      } }
```

4.5.6.7 Provider Text

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

4.5.6.7.1 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
```

4.5.6.7.2 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."      }    }  ] }
```

4.5.6.8 Content

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

4.5.6.8.1 Content request

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

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

4.5.6.8.2 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",    }  } }
```

4.5.6.9 Verification

See for more information on the source of these fields.



4.5.6.9.1 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" } } }
```

4.5.6.9.2 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 } } }
```

4.5.6.9.3 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" } } }
```

4.5.6.9.4 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 } } }
```

4.5.6.10 Cookies

4.5.6.10.1 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" } } ] }
```

4.5.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

4.5.6.11 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.

4.5.6.11.1 Documentation request

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

4.5.6.11.2 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"
] }
```

4.5.6.12 i18 internationalization

✓ *Retrieves i18next json based on requested language*

4.5.6.12.1 i18 request

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



5 Big

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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI
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	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Session	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"



field name	type	constrains	description	example
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-h0mLD2zDgf26ptjxSNRiKFILM",    "payment-session-id":
"4b7c0aec-3db3-4cd2-9b22-4d41fb964fdc",    "payment-status": "pending",    "payment-
method": "creditcard",    "voucher-code": "LEBALLINN10",    "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-h0mLD2zDgf26ptjxSNRiKFILM", "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 [GetProviderTexts] 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						

5.2 Edge Service

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

5.2.1 TODO

5.2.1.1 TODO version 1.0

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

5.2.1.2 TODO version 1.0-next

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

5.2.2 Test environment and credentials

TODO

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

5.2.3 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. Therefore 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.

5.2.4 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.

5.2.5 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.

5.2.6 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].

5.2.6.1 Login

5.2.6.1.1 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}	MzAwMDAwMTY6dG9wLXNlY3Jl dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

5.2.6.1.2 Login response

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

5.2.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

5.2.6.2 Logout

5.2.6.2.1 Logout request

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

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

5.2.6.2.2 Logout response

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

5.2.6.3 Voucher list

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



5.2.6.3.1 Voucher list request

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

5.2.6.3.2 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"
          }
        }
      }
    }
  ]
}
```

5.2.6.4 Voucher detail

5.2.6.4.1 Voucher detail request

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

5.2.6.4.2 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"
        }
      }
    }
  }
}
```

5.2.6.5 Voucher order

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

5.2.6.5.1 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



5.2.6.5.2 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"      }      } }
```

5.2.6.6 Voucher order detail

5.2.6.6.1 Voucher order detail request

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

5.2.6.6.2 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"      }      } }
```

5.2.6.7 Provider Text

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

5.2.6.7.1 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
```

5.2.6.7.2 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."      }    }  ] }
```

5.2.6.8 Content

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

5.2.6.8.1 Content request

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

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

5.2.6.8.2 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",    }  } }
```

5.2.6.9 Verification

See for more information on the source of these fields.



5.2.6.9.1 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" } } }
```

5.2.6.9.2 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 } } }
```

5.2.6.9.3 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" } } }
```

5.2.6.9.4 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 } } }
```

5.2.6.10 Cookies

5.2.6.10.1 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" } } ] }
```

5.2.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

5.2.6.11 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.

5.2.6.11.1 Documentation request

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

5.2.6.11.2 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"
  ]
}

```

5.2.6.12 i18 internationalization

- ✓ Retrieves i18next json based on requested language

5.2.6.12.1 i18 request

```

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

```

5.3 Edge Service

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

5.3.1 TODO

5.3.1.1 TODO version 1.0

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

5.3.1.2 TODO version 1.0-next

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

5.3.2 Test environment and credentials

TODO

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

5.3.3 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. Therefore 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 unregister on shutdown.

5.3.4 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.](#)

5.3.5 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.

5.3.6 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].

5.3.6.1 Login

5.3.6.1.1 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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

5.3.6.1.2 Login response

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

5.3.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

5.3.6.2 Logout

5.3.6.2.1 Logout request

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

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

5.3.6.2.2 Logout response

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

5.3.6.3 Voucher list

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



5.3.6.3.1 Voucher list request

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

5.3.6.3.2 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"
          }
        }
      }
    ]
  }
}
```

5.3.6.4 Voucher detail

5.3.6.4.1 Voucher detail request

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

5.3.6.4.2 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"
        }
      }
    }
  }
}
```

5.3.6.5 Voucher order

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

5.3.6.5.1 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



5.3.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

5.3.6.6 Voucher order detail

5.3.6.6.1 Voucher order detail request

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

5.3.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

5.3.6.7 Provider Text

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

5.3.6.7.1 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
```

5.3.6.7.2 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."      }    }  ] }
```

5.3.6.8 Content

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

5.3.6.8.1 Content request

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

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

5.3.6.8.2 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",    }  } }
```

5.3.6.9 Verification

See for more information on the source of these fields.



5.3.6.9.1 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" } } }
```

5.3.6.9.2 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 } } }
```

5.3.6.9.3 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" } } }
```

5.3.6.9.4 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 } } }
```

5.3.6.10 Cookies

5.3.6.10.1 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" } } ] }
```

5.3.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

5.3.6.11 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.

5.3.6.11.1 Documentation request

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

5.3.6.11.2 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"
  ]
}

```

5.3.6.12 *i18n internationalization*

- ✓ *Retrieves i18next json based on requested language*

5.3.6.12.1 *i18n request*

```

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

```

5.4 Edge Service

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

5.4.1 *TODO*

5.4.1.1 *TODO version 1.0*

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

5.4.1.2 *TODO version 1.0-next*

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

5.4.2 *Test environment and credentials*

TODO

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

5.4.3 *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.

5.4.4 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.](#)

5.4.5 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.

5.4.6 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].

5.4.6.1 Login

5.4.6.1.1 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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

5.4.6.1.2 Login response

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

5.4.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

5.4.6.2 Logout

5.4.6.2.1 Logout request

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

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

5.4.6.2.2 Logout response

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

5.4.6.3 Voucher list

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



5.4.6.3.1 Voucher list request

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

5.4.6.3.2 Voucher list response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": [
    {
      "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": "f634f09f-9a06-4297-a326-0d703938b702"
    },
    {
      "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"
          }
        }
      },
      "id": "a3473e20-19b4-44fe-8389-67a511854759"
    }
  ]
}
```

5.4.6.4 Voucher detail

5.4.6.4.1 Voucher detail request

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

5.4.6.4.2 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"
        }
      }
    }
  }
}
```

5.4.6.5 Voucher order

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

5.4.6.5.1 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



5.4.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"    }  } }
```

5.4.6.6 Voucher order detail

5.4.6.6.1 Voucher order detail request

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

5.4.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"    }  } }
```

5.4.6.7 Provider Text

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

5.4.6.7.1 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
```

5.4.6.7.2 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."    }    }  ] }
```

5.4.6.8 Content

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

5.4.6.8.1 Content request

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

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

5.4.6.8.2 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",    }    }  }
```

5.4.6.9 Verification

See for more information on the source of these fields.



5.4.6.9.1 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" } } }
```

5.4.6.9.2 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 } } }
```

5.4.6.9.3 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" } } }
```

5.4.6.9.4 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 } } }
```

5.4.6.10 Cookies

5.4.6.10.1 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" } } ] }
```

5.4.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

5.4.6.11 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.

5.4.6.11.1 Documentation request

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

5.4.6.11.2 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"
] }
```

5.4.6.12 *i18 internationalization*

- ✓ *Retrieves i18next json based on requested language*

5.4.6.12.1 *i18 request*

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

5.5 Edge Service

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

5.5.1 *TODO*

5.5.1.1 *TODO version 1.0*

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

5.5.1.2 *TODO version 1.0-next*

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

5.5.2 *Test environment and credentials*

TODO

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

5.5.3 *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.

5.5.4 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.](#)

5.5.5 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.

5.5.6 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].

5.5.6.1 Login

5.5.6.1.1 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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

5.5.6.1.2 Login response

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

5.5.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

5.5.6.2 Logout

5.5.6.2.1 Logout request

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

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

5.5.6.2.2 Logout response

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

5.5.6.3 Voucher list

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



5.5.6.3.1 Voucher list request

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

5.5.6.3.2 Voucher list response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": [
    {
      "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": "f634f09f-9a06-4297-a326-0d703938b702"
    },
    {
      "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"
          }
        }
      },
      "id": "a3473e20-19b4-44fe-8389-67a511854759"
    }
  ]
}
```

5.5.6.4 Voucher detail

5.5.6.4.1 Voucher detail request

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

5.5.6.4.2 Voucher detail response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": {
    "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": "f634f09f-9a06-4297-a326-0d703938b702"
  }
}
```

5.5.6.5 Voucher order

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

5.5.6.5.1 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



5.5.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

5.5.6.6 Voucher order detail

5.5.6.6.1 Voucher order detail request

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

5.5.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

5.5.6.7 Provider Text

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

5.5.6.7.1 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
```

5.5.6.7.2 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."    }    }  ] }
```

5.5.6.8 Content

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

5.5.6.8.1 Content request

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

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

5.5.6.8.2 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",    }  } }
```

5.5.6.9 Verification

See for more information on the source of these fields.



5.5.6.9.1 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" } } }
```

5.5.6.9.2 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 } } }
```

5.5.6.9.3 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" } } }
```

5.5.6.9.4 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 } } }
```

5.5.6.10 Cookies

5.5.6.10.1 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" } } ] }
```

5.5.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

5.5.6.11 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.

5.5.6.11.1 Documentation request

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

5.5.6.11.2 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"
] }
```

5.5.6.12 i18 internationalization

✓ *Retrieves i18next json based on requested language*

5.5.6.12.1 i18 request

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



6 Big

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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI
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	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Session	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"



field name	type	constrains	description	example
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

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-h0mLD2zDgf26ptjxSNRiKFILM",      "payment-session-id": "4b7c0aec-3db3-4cd2-9b22-4d41fb964fdc",      "payment-status": "pending",      "payment-method": "creditcard",      "voucher-code": "LEBALLINN10",      "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"    }  } }
```

6.1.6.6 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-h0mLD2zDgf26ptjxSNRiKFILM", "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 [GetProviderTexts] 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 i18n internationalization

- ✓ Retrieves i18next json based on requested language

6.1.6.12.1 i18n request

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

6.2 Edge Service

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

6.2.1 TODO

6.2.1.1 TODO version 1.0

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

6.2.1.2 TODO version 1.0-next

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

6.2.2 Test environment and credentials

TODO

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

6.2.3 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. Therefore 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.

6.2.4 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.

6.2.5 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.

6.2.6 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].

6.2.6.1 Login

6.2.6.1.1 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}	MzAwMDAwMTY6dG9wLXNlY3Jl dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

6.2.6.1.2 Login response

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

6.2.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

6.2.6.2 Logout

6.2.6.2.1 Logout request

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

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

6.2.6.2.2 Logout response

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

6.2.6.3 Voucher list

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



6.2.6.3.1 Voucher list request

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

6.2.6.3.2 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"
          }
        }
      }
    ]
  }
}
```

6.2.6.4 Voucher detail

6.2.6.4.1 Voucher detail request

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

6.2.6.4.2 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"
        }
      }
    }
  }
}
```

6.2.6.5 Voucher order

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

6.2.6.5.1 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



6.2.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

6.2.6.6 Voucher order detail

6.2.6.6.1 Voucher order detail request

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

6.2.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

6.2.6.7 Provider Text

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

6.2.6.7.1 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
```

6.2.6.7.2 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."    }    }  ] }
```

6.2.6.8 Content

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

6.2.6.8.1 Content request

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

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

6.2.6.8.2 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",    }  } }
```

6.2.6.9 Verification

See for more information on the source of these fields.



6.2.6.9.1 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" } } }
```

6.2.6.9.2 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 } } }
```

6.2.6.9.3 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" } } }
```

6.2.6.9.4 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 } } }
```

6.2.6.10 Cookies

6.2.6.10.1 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" } } ] }
```

6.2.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

6.2.6.11 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.

6.2.6.11.1 Documentation request

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

6.2.6.11.2 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"
] }
```

6.2.6.12 i18 internationalization

- ✓ Retrieves i18next json based on requested language

6.2.6.12.1 i18 request

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

6.3 Edge Service

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

6.3.1 TODO

6.3.1.1 TODO version 1.0

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

6.3.1.2 TODO version 1.0-next

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

6.3.2 Test environment and credentials

TODO

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

6.3.3 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.

6.3.4 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.](#)

6.3.5 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.

6.3.6 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].

6.3.6.1 Login

6.3.6.1.1 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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

6.3.6.1.2 Login response

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

6.3.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

6.3.6.2 Logout

6.3.6.2.1 Logout request

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

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

6.3.6.2.2 Logout response

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

6.3.6.3 Voucher list

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



6.3.6.3.1 Voucher list request

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

6.3.6.3.2 Voucher list response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": [
    {
      "type": "voucher",
      "currency": "EUR",
      "kind": 0,
      "name": "Lebara All-in NL 10 EUR",
      "provider-name": "Lebara",
      "price-total": 11.99,
      "payment-methods": {
        "wirecard-creditcard": {
          "name": "Credit Card",
          "provider-name": "Wirecard",
          "image-url": "https://wirecard.com/logo.png"
        },
        "wirecard-paypal": {
          "name": "PayPal",
          "provider-name": "Wirecard"
        }
      },
      "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-tax": 0.00,
        "price-fee": 1.99,
        "name": "Credit Card",
        "provider-name": "Wirecard",
        "image-url": "https://wirecard.com/logo.png"
      }
    },
    {
      "type": "voucher",
      "currency": "EUR",
      "kind": 0,
      "name": "Lebara All-in NL 15 EUR",
      "provider-name": "Lebara",
      "price-total": 16.99,
      "payment-methods": {
        "wirecard-creditcard": {
          "name": "Credit Card",
          "provider-name": "Wirecard",
          "image-url": "https://wirecard.com/logo.png"
        },
        "wirecard-paypal": {
          "name": "PayPal",
          "provider-name": "Wirecard"
        }
      },
      "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-tax": 0.00,
        "price-fee": 1.99,
        "name": "Credit Card",
        "provider-name": "Wirecard",
        "image-url": "https://wirecard.com/logo.png"
      }
    }
  ]
}
```

6.3.6.4 Voucher detail

6.3.6.4.1 Voucher detail request

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

6.3.6.4.2 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"
        }
      }
    }
  }
}
```

6.3.6.5 Voucher order

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

6.3.6.5.1 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



6.3.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

6.3.6.6 Voucher order detail

6.3.6.6.1 Voucher order detail request

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

6.3.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

6.3.6.7 Provider Text

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

6.3.6.7.1 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
```

6.3.6.7.2 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."      }    }  ] }
```

6.3.6.8 Content

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

6.3.6.8.1 Content request

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

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

6.3.6.8.2 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",    }  } }
```

6.3.6.9 Verification

See for more information on the source of these fields.



6.3.6.9.1 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" } } }
```

6.3.6.9.2 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 } } }
```

6.3.6.9.3 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" } } }
```

6.3.6.9.4 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 } } }
```

6.3.6.10 Cookies

6.3.6.10.1 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" } } ] }
```

6.3.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

6.3.6.11 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.

6.3.6.11.1 Documentation request

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

6.3.6.11.2 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"
] }
```

6.3.6.12 *i18 internationalization*

- ✓ *Retrieves i18next json based on requested language*

6.3.6.12.1 *i18 request*

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

6.4 Edge Service

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

6.4.1 *TODO*

6.4.1.1 *TODO version 1.0*

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

6.4.1.2 *TODO version 1.0-next*

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

6.4.2 *Test environment and credentials*

TODO

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

6.4.3 *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.

6.4.4 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.](#)

6.4.5 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.

6.4.6 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].

6.4.6.1 Login

6.4.6.1.1 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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

6.4.6.1.2 Login response

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

6.4.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

6.4.6.2 Logout

6.4.6.2.1 Logout request

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

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

6.4.6.2.2 Logout response

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

6.4.6.3 Voucher list

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



6.4.6.3.1 Voucher list request

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

6.4.6.3.2 Voucher list response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": [
    {
      "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": "f634f09f-9a06-4297-a326-0d703938b702"
    },
    {
      "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"
          }
        }
      },
      "id": "a3473e20-19b4-44fe-8389-67a511854759"
    }
  ]
}
```

6.4.6.4 Voucher detail

6.4.6.4.1 Voucher detail request

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

6.4.6.4.2 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"
        }
      }
    }
  }
}
```

6.4.6.5 Voucher order

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

6.4.6.5.1 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



6.4.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

6.4.6.6 Voucher order detail

6.4.6.6.1 Voucher order detail request

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

6.4.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

6.4.6.7 Provider Text

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

6.4.6.7.1 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
```

6.4.6.7.2 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."      }    }  ] }
```

6.4.6.8 Content

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

6.4.6.8.1 Content request

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

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

6.4.6.8.2 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",    }  } }
```

6.4.6.9 Verification

See for more information on the source of these fields.



6.4.6.9.1 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" } } }
```

6.4.6.9.2 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 } } }
```

6.4.6.9.3 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" } } }
```

6.4.6.9.4 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 } } }
```

6.4.6.10 Cookies

6.4.6.10.1 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" } } ] }
```

6.4.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

6.4.6.11 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.

6.4.6.11.1 Documentation request

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

6.4.6.11.2 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"
] }
```

6.4.6.12 i18 internationalization

- ✓ *Retrieves i18next json based on requested language*

6.4.6.12.1 i18 request

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

6.5 Edge Service

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

6.5.1 TODO

6.5.1.1 TODO version 1.0

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

6.5.1.2 TODO version 1.0-next

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

6.5.2 Test environment and credentials

TODO

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

6.5.3 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.

6.5.4 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.](#)

6.5.5 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.

6.5.6 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].

6.5.6.1 Login

6.5.6.1.1 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}	MzAwMDAwMTY6dG9wLXNIY3JI dC1wYXNzcGhyYXNI



```
POST /api/rest/v1/auth/login HTTP/1.1 Authorization: Basic ${ICP_SHOP_DIGEST} Accept: application/vnd.api+json
```

6.5.6.1.2 Login response

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

6.5.6.1.3 Cookies

Name	Value	Domain	Path	Expires	HttpOnly	Secure
SESSION	MmQ5ZjkxO GltMTM4My0 0M2E0LWI1M DYtYmUwZjY wMzJiZDY5	master.env.a pi.shop1.roc ks	/	Seassion	true	false

6.5.6.2 Logout

6.5.6.2.1 Logout request

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

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

6.5.6.2.2 Logout response

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

6.5.6.3 Voucher list

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



6.5.6.3.1 Voucher list request

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

6.5.6.3.2 Voucher list response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": [
    {
      "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": "f634f09f-9a06-4297-a326-0d703938b702"
    },
    {
      "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"
          }
        }
      },
      "id": "a3473e20-19b4-44fe-8389-67a511854759"
    }
  ]
}
```

6.5.6.4 Voucher detail

6.5.6.4.1 Voucher detail request

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

6.5.6.4.2 Voucher detail response

```
HTTP/1.1 200 OK Content-Type: application/vnd.api+json
{
  "data": {
    "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": "f634f09f-9a06-4297-a326-0d703938b702"
  }
}
```

6.5.6.5 Voucher order

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

6.5.6.5.1 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



6.5.6.5.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

6.5.6.6 Voucher order detail

6.5.6.6.1 Voucher order detail request

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

6.5.6.6.2 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-h0mLD2zDgf26ptjxSNRiKFiLM",      "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"      }      } }
```

6.5.6.7 Provider Text

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

6.5.6.7.1 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
```

6.5.6.7.2 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."    }    }  ] }
```

6.5.6.8 Content

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

6.5.6.8.1 Content request

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

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

6.5.6.8.2 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",    }  } }
```

6.5.6.9 Verification

See for more information on the source of these fields.



6.5.6.9.1 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" } } }
```

6.5.6.9.2 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 } } }
```

6.5.6.9.3 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" } } }
```

6.5.6.9.4 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 } } }
```

6.5.6.10 Cookies

6.5.6.10.1 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" } } ] }
```

6.5.6.10.2 Cookies response

```
HTTP/1.1 200 OK
```

6.5.6.11 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.

6.5.6.11.1 Documentation request

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

6.5.6.11.2 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"
] }
```

6.5.6.12 i18 internationalization

✓ *Retrieves i18next json based on requested language*

6.5.6.12.1 i18 request

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



We enable docx-as-code



Figure 1: Last Page